



---

CENTRO "PAULA SOUZA"

FACULDADE DE TECNOLOGIA DE AMERICANA – "MINISTRO RALPH BIASI"  
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Abner Ribeiro Da Silva  
Elcio Roberto Cardoso Pereira  
Mateus Antonio Rodrigues Lopes  
Vitor Marcolin

**Desenvolvimento de um aplicativo de agenda para estudantes: School  
Schedule**

AMERICANA, SP

2022



---

CENTRO “PAULA SOUZA”

FACULDADE DE TECNOLOGIA DE AMERICANA – “MINISTRO RALPH BIASI”  
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Abner Ribeiro Da Silva  
Elcio Roberto Cardoso Pereira  
Mateus Antonio Rodrigues Lopes  
Vitor Marcolin

**Desenvolvimento de um aplicativo de agenda para estudantes: School  
Schedule**

Trabalho de Conclusão do Curso Superior de  
Tecnologia em Análise e Desenvolvimento de  
Sistemas, sob a orientação do Prof. MSc.  
Wagner Siqueira Cavalcante.

Área de concentração: Engenharia de Software

AMERICANA, SP

2022

Abner Ribeiro Da Silva  
Elcio Roberto Cardoso Pereira  
Mateus Antonio Rodrigues Lopes  
Vitor Marcolin

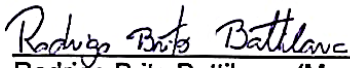
**Desenvolvimento de um aplicativo de agenda para estudantes: *School  
Schedule***

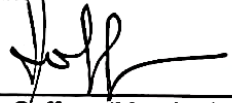
Trabalho de graduação apresentado como exigência parcial para obtenção do título de Tecnólogo em Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas pelo Centro Paula Souza – FATEC Faculdade de Tecnologia de Americana – Ralph Biasi.  
Área de concentração: Engenharia de Software.

Americana, 06 de dezembro de 2022

**Banca Examinadora:**

  
\_\_\_\_\_  
Wagner Siqueira Cavalcante (Presidente)  
Mestre  
Fatec Americana

  
\_\_\_\_\_  
Rodrigo Brito Battilana (Membro)  
Mestre  
Fatec Americana

  
\_\_\_\_\_  
Renato Kraide Soffner (Membro)  
Doutor  
Fatec Americana

## SUMÁRIO

1.	INTRODUÇÃO .....	12
2.	ANÁLISE DOS CONCORRENTES .....	14
3.	PROJETO DO SISTEMA .....	16
3.1.	Levantamento de Requisitos .....	16
3.1.1.	Requisitos funcionais .....	16
3.1.2.	Requisitos não funcionais .....	17
3.2.	Recursos e Ferramentas .....	18
3.3.	Metodologia.....	19
3.4.	Etapas de desenvolvimento .....	20
3.4.1.	<i>Sprint 1</i> .....	22
3.4.2.	<i>Sprint 2</i> .....	24
3.4.3.	<i>Sprint 3</i> .....	26
3.4.4.	<i>Sprint 4</i> .....	28
3.4.5.	<i>Sprint 5</i> .....	30
3.4.6.	<i>Sprint 6</i> .....	31
3.4.7.	<i>Sprint 7</i> .....	34
3.4.8.	<i>Sprint 8</i> .....	36
3.4.9.	<i>Sprint 9</i> .....	38
3.4.10.	<i>Sprint 10</i> .....	40
3.4.11.	<i>Sprint 11</i> .....	41

3.4.12. <i>Sprint 12</i> .....	43
3.4.13. <i>Sprint 13</i> .....	46
3.4.14. <i>Sprint 14</i> .....	47
3.5. Desenvolvimento do protótipo de alta fidelidade .....	49
4. MODELAGEM .....	86
4.1. UML: Diagramas de Casos de Uso .....	86
4.1.1. Documentação dos Casos de Uso .....	90
4.2. UML: Diagrama de Classes .....	96
4.3. UML: Diagramas de Sequência .....	97
4.4. Banco de Dados .....	103
4.5. Diagrama De Entidade E Relacionamento (DER) .....	104
4.6. Estruturação <i>SQL</i> .....	105
4.7. Dicionário De Dados .....	106
5. CONSIDERAÇÕES FINAIS .....	112
REFERÊNCIAS .....	114

## LISTA DE FIGURAS

Figura 1 – Gráfico de <i>Burndown</i> da entrega 1.....	23
Figura 2 – Gráfico de <i>Burndown</i> da entrega 2.....	25
Figura 3 - Gráfico de <i>Burndown</i> da entrega 3 .....	27
Figura 4 - Gráfico de <i>Burndown</i> da entrega 4 .....	29
Figura 5 – Gráfico de <i>Burndown</i> da entrega 5.....	31
Figura 6 - Gráfico de <i>Burndown</i> da entrega 6 .....	33
Figura 7 - Gráfico de <i>Burndown</i> da entrega 7 .....	35
Figura 8 – Gráfico de <i>Burndown</i> da entrega 8.....	37
Figura 9 – Gráfico de <i>Burndown</i> da entrega 9.....	39
Figura 10 – Gráfico de <i>Burndown</i> da entrega 10.....	41
Figura 11 – Gráfico de <i>Burndown</i> da entrega 11.....	43
Figura 12 – Gráfico de <i>Burndown</i> da entrega 12.....	45
Figura 13 - Gráfico de <i>Burndown</i> da entrega 13 .....	47
Figura 14 – Gráfico de <i>Burndown</i> da entrega 14.....	49
Figura 15 – Diagrama de estados (mapa das telas).....	50
Figura 16 - Captura da <i>Splash Screen</i> .....	51
Figura 17 – Captura da Tela Inicial .....	52
Figura 18 – Captura da tela Cadastro .....	53
Figura 19 - Captura da Tela de <i>Login</i> .....	54
Figura 20 - Captura da Tela <i>Home</i> .....	55
Figura 21 - Captura da Tela Configurações .....	56
Figura 22 - Captura da tela Editar perfil .....	57
Figura 23 - Captura da tela Alterar senha .....	58
Figura 24 - Captura da Tela de Disciplinas .....	59
Figura 25 - Captura da Tela adicionar disciplina .....	60
Figura 26 - Captura da tela Disciplina .....	61
Figura 27 - Captura da tela Editar Disciplina .....	62
Figura 28 - Captura da tela Deletar Disciplina.....	63
Figura 29 - Captura da tela Notas. ....	64
Figura 30 - Captura da tela Adicionar nota.....	65
Figura 31 - Captura da tela Dropdown das notas.....	66

Figura 32 - Captura da tela Notas .....	67
Figura 33 - Captura da tela Editar nota .....	68
Figura 34 - Captura da tela com a opção para deletar nota .....	69
Figura 35 - Captura da tela Ausências .....	70
Figura 36 - Captura da tela “Adicionar ausência” .....	71
Figura 37 - Captura da tela <i>Dropdown</i> de ausências .....	72
Figura 38 - Captura da ausência .....	73
Figura 39 - Captura da tela Editar ausência .....	74
Figura 40 - Captura da tela de opção para deletar ausência, na tela “Ausências” .....	75
Figura 41 - Captura da tela Horários .....	76
Figura 42 - Captura da tela adicionar horário .....	77
Figura 43 - Captura da tela Horários .....	78
Figura 44 - Captura da tela com a opção para editar horário, da tela Horários .....	79
Figura 45 - Captura da Deletar horário .....	80
Figura 46 - Captura da tela Agenda .....	81
Figura 47 - Captura da tela com a opção para adicionar evento na tela “Agenda” .....	82
Figura 48 - Captura da tela com a opção evento na tela Agenda .....	83
Figura 49 - Captura da tela editar evento .....	84
Figura 50 - Captura da deletar evento .....	85
Figura 51 – Diagrama de caso de uso do cadastro e login do usuário .....	87
Figura 52 – Diagrama de caso de uso do CRUD de disciplinas .....	87
Figura 53 – Diagrama de Caso de Uso do CRUD de Notas .....	88
Figura 54 - Diagrama de Caso de Uso do CRUD de Horários .....	88
Figura 55 – Diagrama de Caso de Uso do CRUD de Eventos .....	89
Figura 56 – Diagrama de Caso de Uso do CRUD de Ausências .....	89
Figura 57 – Diagrama de Classes do aplicativo .....	97
Figura 58 – Diagrama de sequência do cadastro e login do usuário .....	98
Figura 59 – Diagrama de sequência de manter disciplina .....	99
Figura 60 – Diagrama a de Sequência de manter nota .....	100
Figura 61 – Diagrama de Sequência de manter horário .....	101
Figura 62 – Diagrama de Sequência de manter evento .....	102
Figura 63 – Diagrama de Sequência de manter ausência .....	103
Figura 64 - Representação o Diagrama de Entidade e Relacionamento do aplicativo .....	104

Figura 65 - Representação do objeto Disciplina.....	106
---	-----



## LISTA DE TABELAS

Tabela 1 - Comparativo de funcionalidades da aplicação <i>School Schedule</i> em relação aos concorrentes.....	15
Tabela 2 - Requisitos funcionais do projeto. ....	16
Tabela 3 – Requisitos não funcionais do projeto.....	17
Tabela 4 - Planejamento realizado para primeira entrega.....	22
Tabela 5 – Planejamento realizado para segunda entrega.....	24
Tabela 6 – Planejamento realizado para terceira entrega.....	26
Tabela 7 – Planejamento realizado para quarta entrega.....	28
Tabela 8 – Planejamento realizado para quinta entrega.....	30
Tabela 9 – Planejamento realizado para sexta entrega.....	32
Tabela 10 – Planejamento realizado para sétima entrega.....	34
Tabela 11 - Planejamento realizado para oitava entrega.....	36
Tabela 12 - Planejamento realizado para nona entrega.....	38
Tabela 13 - Planejamento realizado para décima entrega.....	40
Tabela 14 - Planejamento realizado para décima primeira entrega.....	42
Tabela 15 - Planejamento realizado para décima segunda entrega.....	44
Tabela 16 - Planejamento realizado para décima terceira entrega.....	46
Tabela 17 - Planejamento realizado para décima quarta entrega.....	48
Tabela 18 – Caso de uso “Entrar no Sistema / Cadastrar Usuário”. ....	90
Tabela 19 – Caso de uso “Inserir / Editar / Deletar Disciplina”. ....	90
Tabela 20 – Caso de uso “Inserir / Modificar / Deletar Nota”. ....	92
Tabela 21 – Caso de uso “Inserir / Modificar Horários”. ....	93
Tabela 22 – Caso de uso “Inserir / Modificar Eventos”.....	94
Tabela 23 – Caso de uso “Inserir / Modificar / Deletar Ausências”.....	95
Tabela 24 - Dicionário de dados da entidade usuário.....	107
Tabela 25 - Dicionário de dados da entidade disciplina.....	108
Tabela 26 - Dicionário de dados da entidade ausência.....	109
Tabela 27 - Dicionário de dados da entidade nota.....	110
Tabela 28- Dicionário de dados da entidade evento.....	110
Tabela 29 - Dicionário de dados da entidade Horário.....	111

## LISTA DE SIGLAS E ABREVIATURAS

ABED	Associação Brasileira de Educação a Distância
Saresp	Avaliação de Rendimento Escolar do Estado de São Paulo
CRUD	<i>Create Read Update and Delete</i> (Criar Ler Atualizar e Deletar)
MVC	<i>Model View Controller</i> (Arquitetura Modelo-Visão-Controlle)
SQL	<i>Structured Query Language</i> (Linguagem de Consulta Estruturada)
UML	<i>Unified Modeling Language</i> (Linguagem de Modelagem Unificada)

## RESUMO

Por intervenção da pandemia de covid-19, estudantes de todos os níveis tiveram que estudar remotamente para a segurança dos mesmos. Em função disso, muitos estudantes passaram dificuldades para estabelecer uma organização nos seus estudos, levando a uma queda de desempenho, isto fica evidente através da performance dos alunos do ensino médio do Estado de São Paulo, que tiveram o pior desempenho da história na Avaliação de Rendimento Escolar do Estado de São Paulo. Em razão dos problemas apresentados, foi pensado no desenvolvimento deste trabalho. *School Schedule* é um aplicativo de agenda focado em estudantes, com o objetivo de auxiliá-los a se organizarem na vida acadêmica por meio de uma interface simples e intuitiva, os usuários serão capazes de criar disciplinas, adicionar notas, controlar ausências e gerenciar eventos. Para o desenvolvimento deste aplicativo inicialmente foi delineado o escopo e requisitos dos problemas a serem solucionados, fez-se uma pesquisa de mercado, um protótipo de alta fidelidade e foi desenvolvido o projeto utilizando a metodologia *scrum* que torna os processos mais simples e claros, pois mantém registros visíveis sobre o andamento de todas as etapas. Como linguagem de programação foi utilizado o *Dart* e como *framework Flutter*, que atualmente é um dos principais *frameworks* para o desenvolvimento de aplicações mobile. Para fazer a criação e autenticação de contas foi empregado o *Firestore* e para banco de dados o *SQLite*. Ao fim do desenvolvimento, ficou evidente que se utilizar um aplicativo para gerenciar a vida acadêmica auxilia na organização dos estudos, tornando-se possível melhorar a performance no âmbito escolar. Portanto, o aplicativo oferece uma assistência para estudantes terem mais controle do seu desempenho nos estudos.

**Palavras Chave:** Desempenho; Estudantes; Estudos; Organização.

## ABSTRACT

Due to the covid-19 pandemic, students at all levels had to take classes remotely for their safety. As a result, many students had difficulties in establishing an organization in their studies, leading to a drop in performance, this is evident through the performance of high school students in the State of São Paulo, who had the worst performance in history in the Evaluation of School Performance of the State of São Paulo. The development of this work was thought of because of these problems presented. School Schedule is a calendar application focused on students, with the objective of helping them to organize their academic life through a simple and intuitive interface. Users will be able to create the subjects they are studying, add their grades, control their absences and manage events. To develop this application, the scope and requirements of the problems to be solved were initially outlined, market research was done, a high-fidelity prototype was made, and the project was developed using the scrum methodology, which makes the processes simpler and clearer because it keeps visible records about the progress of all stages. The programming language used was Dart and the framework Flutter, which is currently one of the main frameworks for the development of mobile applications. Firebase was used to create and authenticate accounts, and SQLite was used for the database. At the end of the development, it became evident that using an application to manage academic life helps to organize studies, making it possible to improve performance at school. Therefore, the application offers assistance for students to have more control over their study performance.

**Keywords:** Organize; Students; Studies; Performance.

## 1. INTRODUÇÃO

Com o advento da pandemia de Covid-19 iniciada em 2019, todas as escolas foram obrigadas a implementar medidas rigorosas de isolamento social para evitar a disseminação do vírus, as aulas presenciais foram interrompidas e passaram a ser lecionadas remotamente. Nesse cenário, os alunos foram impactados com essa mudança radical, muitos não conseguiram se adaptar a esse novo modelo de ensino tendo muitas adversidades no dia a dia, dentre essas adversidades temos falta de organização e planejamento, é possível evidenciar isso através de uma pesquisa divulgada pelo Estadão em 2020, no qual a Associação Brasileira de Educação a Distância (ABED) realizou, com uma amostra de 5.580 participantes -- entre estudantes, professores, pais e/ou responsáveis e dirigentes de instituições de ensino públicas e privadas do país, sobre as atividades remotas na educação, 67% dos alunos relataram de dificuldades em estabelecer uma organização e rotina diária de estudos.

Além disso, de acordo com os resultados divulgados pelo G1 do SARESP 2021 (Sistema de avaliação do Rendimento Escolas do Estado de São Paulo), os alunos do ensino médio tiveram o pior desempenho da história, tendo como as principais disciplinas afetadas com alunos de baixa performance respectivamente: matemática e português, cerca de 96% dos alunos na disciplina de matemática e 76% na disciplina de língua portuguesa. Estes dados evidenciam os impactos negativos causados pelas aulas remotas que contribuíram para que os alunos diminuíssem o seu rendimento nos estudos, causando um grande retrocesso no ciclo de aprendizado.

Levando em conta os dados apresentados foi pensado no desenvolvimento deste trabalho que tem como intuito realizar o desenvolvimento de um aplicativo de agenda, tendo como principal finalidade auxiliar a vida acadêmica dos estudantes, com funcionalidades que buscam facilitar a organização nos estudos. Através de uma interface intuitiva e simples, os usuários serão capazes de criar uma conta utilizando um e-mail e senha, incluir informações de disciplinas, notas, ausências e provas. A utilização do aplicativo auxiliará na produtividade, redução de estresse diário nos estudos e economia de tempo.

O trabalho está organizado em cinco capítulos, sendo que o Capítulo 2 apresenta a análise de concorrentes, Capítulo 3 descreve o projeto do sistema,

Capítulo 4 a modelagem que apresenta os diagramas e por fim Capítulo 5 que são as considerações finais.

## 2. ANÁLISE DOS CONCORRENTES

A fim de comparar e entender melhor o sistema, softwares similares ao que se pretende desenvolver foram analisados, e o que se verificou foi que atualmente existem vários aplicativos de agenda focados em estudantes, cada um com diferentes funcionalidades. Foram selecionadas quatro aplicações mais populares e mais bem avaliadas pelos usuários da *Play Store*, que são:

- **Agenda do Estudante Pro:** aplicação que contém informações importantes sobre provas, trabalhos, compromissos e horários de aulas que estarão sempre disponíveis no *smartphone* para eventuais novos agendamentos e consultas, onde quer que o estudante esteja. Um recurso muito útil são os alarmes e notificações de eventos (lembretes) para ajudar a não se esquecer de atividades importantes. Aplicação com nota 4,8 na *Google Play* (GOOGLE PLAY, 2022a).
- **Agenda do Universitário:** é um aplicativo de gerenciamento estudantil, focado no aluno universitário, para o controle das disciplinas, notas, faltas, fotos, grade de horários, além do controle de eventos, sejam eles provas, atividades, trabalhos, devolução de livros ou qualquer outro que o universitário desejar se lembrar posteriormente, já que é possível agendar notificações do evento no horário que quiser. Aplicação com nota 4,9 na *Google Play* (GOOGLE PLAY, 2022b).
- **Notas U:** é uma mochila cheia de ferramentas com uma interface bonita e intuitiva, rápida e fácil de usar, para organizar a vida acadêmica com estas múltiplas opções e recursos: número de períodos ou cortes, peso (%) de períodos ou cortes, resumo do dia, cálculo rápido, semestres, disciplinas, cores do curso, classificações, objetivo, ausências e mais. Aplicação com nota 4,4 na *Google Play* (GOOGLE PLAY, 2022c).
- **Agenda iScholar:** Este aplicativo foi desenvolvido para que alunos e pais possam consultar notas, faltas, ocorrências, horários, catraca, troca de recados, assim como confirmar presença em eventos da escola, fazer matrículas, realizar pagamentos e acompanhar a situação financeira. Deve-se consultar a escola para verificar se ela dá suporte a este aplicativo, bem como solicitar o acesso. Aplicação com nota 4,2 na *Google Play* (GOOGLE PLAY, 2022d).

Levando estes aspectos em consideração, foi elaborada a Tabela 1 mostrando as principais diferenças do *School Schedule*, em relação aos aplicativos citados anteriormente:

**Tabela 1 - Comparativo de funcionalidades da aplicação *School Schedule* em relação aos concorrentes.**

	<b>Agenda do Estudante Pro</b>	<b>Agenda do Universitário</b>	<b>Notas U</b>	<b>Agenda iScholar</b>	<b>School Shedule</b>
Gratuito	X	X	X		X
Sistema de <i>Login</i>				X	X
Dados salvos localmente			X		X
Cadastro de disciplinas	X	X	X	X	X
Cadastro de notas	X		X	X	X
Cadastro de ausências		X	X	X	X
Agenda	X	X	X	X	X
Horários das aulas		X	X		X
Sem anúncios					X
Customização do perfil					X
Compra dentro do aplicativo	X	X	X	X	

**Fonte: Elaborado pelos autores.**



### 3. PROJETO DO SISTEMA

Este capítulo detalha o processo de desenvolvimento do aplicativo *School Schedule*, incluindo o levantamento de requisitos (funcionais e não funcionais), os recursos e as ferramentas que serão utilizados no decorrer do desenvolvimento e a metodologia empregada.

#### 3.1. Levantamento de Requisitos

A engenharia de requisitos (RE – *Requirements Engineering*) é o processo de descobrir, analisar, documentar e verificar requisitos de um sistema. O procedimento de levantamento de requisitos, quando bem elaborado, garante qualidade, eficiência, confiabilidade e integridade ao *software* desenvolvido.

Um requisito pode ser definido como a descrição das características do sistema, tal como suas restrições operacionais ou serviços fornecidos pelo mesmo. É uma condição ou capacidade que deve ser alcançada ou estar presente em um sistema para satisfazer um contrato, padrão ou especificação. Tradicionalmente, os requisitos são divididos em dois tipos: requisitos funcionais e requisitos não funcionais.

##### 3.1.1. Requisitos funcionais

“Os requisitos funcionais descrevem o que o sistema deve fazer, isto é, definem a funcionalidade desejada do *software*” (SOMMERVILLE, 2007).

A Tabela 2 apresenta os requisitos funcionais deste projeto.

**Tabela 2 - Requisitos funcionais do projeto.**

Identificação	Requisito Funcional	Prioridade
RF001	Cadastro de Disciplina	Essencial
RF002	Controle de Ausências	Essencial
RF003	Controle de Notas	Essencial
RF004	Agenda Unificada	Importante

**Fonte: Elaborado pelos autores.**

- **RF001 Cadastro de Disciplina**

O Usuário do aplicativo precisa cadastrar e modificar as disciplinas que está fazendo, para ter total controle do período atual.

- **RF002 Controle de Ausências**

É necessário que o usuário tenha controle absoluto sobre suas ausências, podendo definir o número máximo de faltas em uma disciplina, assim como suas ausências no decorrer do período letivo.

- **RF003 Controle de Notas**

O Sistema deve ser capaz de aceitar as notas conforme forem necessárias, fornecendo um cálculo dinâmico da média do período, possibilitando ao usuário maior controle da sua situação acadêmica.

- **RF004 Agenda Unificada**

A agenda deve conter todas as aulas e eventos cadastrados, a fim de possibilitar uma visão das obrigações do estudante e auxiliar no planejamento da execução das atividades.

### 3.1.2. Requisitos não funcionais

“Os requisitos não funcionais são aqueles não diretamente relacionados às funções específicas fornecidas pelo sistema” (SOMMERVILLE, 2007).

A Tabela 3 apresenta os requisitos não funcionais deste projeto.

**Tabela 3 – Requisitos não funcionais do projeto.**

<b>Identificação</b>	<b>Requisito não funcional</b>	<b>Categoria</b>	<b>Prioridade</b>
RNF001	Interface Intuitiva	Usabilidade	Essencial
RNF002	Interface Responsiva	Usabilidade	Essencial
RNF003	Manter dados de usuário seguros	Segurança	Essencial
RNF004	Arquitetura MVC	Padrões	Essencial

**Fonte: Elaborado pelos autores.**

- **RNF001 Interface Intuitiva**

O Usuário deve facilmente identificar os menus e submenus, para uma usabilidade fluida e sem muitas complicações

- **RNF002 Interface Responsiva**

O Aplicativo deve rodar em qualquer dispositivo móvel, sem prejudicar a interface, e mantendo suas características

- **RNF003 Manter Dados de Usuários Seguros**

Os dados dos usuários não podem em hipótese alguma serem extraviados, ou vazados, provendo uma maior confiabilidade ao aplicativo

- **RNF004 Arquitetura MVC**

O Software seguirá a arquitetura MVC (*Model View Controller*) facilitando o desenvolvimento e a segregação de funções dentro do código, além de futuras manutenções quando necessárias.

### 3.2. Recursos e Ferramentas

Esta seção contempla as ferramentas de programação e os conceitos utilizados para o desenvolvimento do aplicativo:

- **Android Studio:** ambiente de desenvolvimento integrado (*IDE*) oficial para o desenvolvimento de aplicativos *Android*, baseado no *IntelliJ IDEA*, que oferece um ambiente unificado para o desenvolvimento de aplicativos, com o qual é possível desenvolver, fazer *debugs*, testes e interfaces para *smartphones* e *tablets Android* e dispositivos *Android Wear*, *Android TV* e *Android Auto* (*ANDROID*, 2022).
- **Visual Studio Code:** É um editor de código-fonte autônomo que é executado no *Windows*, *macOS* e *Linux*. Ele também contém diversas extensões para auxiliar e agilizar o desenvolvimento das aplicações, sejam elas para *web*, *desktop* ou *mobile* (*MICROSOFT*, 2022).
- **Dart:** É uma linguagem de programação, fortemente tipada, inicialmente criada pela *Google* em 2011. A missão inicial da *Dart* era substituir o *JavaScript* para desenvolvimento de scripts em páginas *web*. Porém, com a evolução da linguagem e com o passar dos anos, ela hoje pode ser considerada uma linguagem multi-paradigma, embora a linguagem apresente fortes estruturas típicas de linguagens orientadas a objeto (*DART*, 2022).
- **Flutter:** É um kit de ferramentas do *Google* para construir aplicações, nativamente compiladas para *mobile*, *web* e *desktop* a partir de um único

código-base. Utilizando a linguagem *Dart* (que também foi criada pelo *Google*), *Flutter* oferece uma incrível experiência para desenvolvedores e usuários. (*FLUTTER*, 2022)

- **Firestore:** É uma plataforma digital de desenvolvimento de aplicativos lançada pelo *Google*, com a qual é possível criar e expandir aplicativos com muita simplicidade, agilidade e facilidade para *Android*, *iOS* e para a *Web*. No aplicativo *Firestore* foi utilizado para fazer o gerenciamento do cadastro e *login* dos usuários (*FIREBASE*, 2022).
- **SQLite:** É a implementação de um dos bancos de dados locais mais famosos do mundo para o *Flutter*, utilizado através do pacote *SQLite*. Foi utilizado para banco de dados local (*PUB.DEV*, 2022).

### 3.3. Metodologia

SCRUM foi a metodologia ágil aplicada para a Gestão do Projeto e consequente orientação para o seu desenvolvimento, *scrum* é um *framework* utilizado para gestão dinâmica de projetos, sendo muitas vezes aplicado para o desenvolvimento ágil de um *software*. É um processo iterativo e incremental, que possui 3 pilares centrais:

- **TRANSPARÊNCIA** dos processos, dos requisitos de entrega e *status*. Todos os aspectos significativos do processo como um todo devem estar visíveis e alinhados com todos os responsáveis pelos resultados.
- **INSPEÇÃO** constante de tudo o que está sendo feito.
- **ADAPTAÇÃO**, tanto do processo, quanto do produto, que podem sofrer mudanças que necessitem de adaptação. Também é importante adaptar o *SCRUM* para a realidade e cultura da organização.

Descrevem-se, também, papéis desempenhados por cada membro:

- **SCRUM MASTER:** responsável por ajudar todo o time a entender e manter vivos os princípios e práticas do *SCRUM* no dia a dia (neste projeto, quem assume este papel é Vitor Marcolin).
- **PRODUCT OWNER:** responsável pelos poderes de liderança sobre o produto. Decide quais recursos serão construídos e a ordem de prioridade (este papel é desempenhado por Mateus Antonio Rodrigues Lopes neste projeto).

- **TEAM:** equipe desenvolvedora do projeto. É o time que define como e quais as coisas serão feitas e quantas tarefas são possíveis de entregar (neste projeto, Abner Ribeiro e Elcio Pereira são os principais, porém todos contribuem).

### 3.4. Etapas de desenvolvimento

Para cada entregável foi convencionado realizar a divisão de tarefas de acordo com a similaridade entre elas. Foi decidido que cada entrega teria um prazo máximo de 15 dias. Resumidamente, em cada entrega foi realizado as seguintes tarefas:

- **Primeira Sprint:** Focou-se principalmente em criar os protótipos de baixa e alta fidelidade e das telas do aplicativo. Utilizando o *Figma*, desenvolveu-se o CRUD de disciplinas, notas, ausências, horários e agenda. Também se realizou o desenvolvimento do *front-end* da tela de *login* com o *Google*.
- **Segunda Sprint:** Efetuou-se o desenvolvimento da interface da *home* e da página do usuário, bem como a *splash screen* e também o protótipo de alta fidelidade das telas no modo escuro, usando o *Figma*.
- **Terceira Sprint:** Decidiu-se fazer um *login e-mail* e senha, pois o *login* com *Google* não estava funcionando. Um protótipo de alta fidelidade das telas “entrar” e “cadastre-se” foi realizado no *Figma*. Em seguida, foram criadas as telas no *Flutter*. Também foi feita a tela “disciplinas” e seu *modal* de adicionar. Foram realizados os ajustes e melhoramentos da interface de praticamente todas as telas e, por fim, também se avançou na documentação da interface do usuário e dos requisitos e ferramentas.
- **Quarta Sprint:** Tentou-se fazer o cadastro e *login* padrão funcionar pela segunda vez, validar campos na tela “*login*” e “*cadastrarse*”, adicionar máscara para o *login*, fazer ajustes de interface, melhorar a documentação, CRUD de disciplinas e notas e, por fim, um protótipo de alta fidelidade das telas “*editar perfil*” e “*trocar senha*”.
- **Quinta Sprint:** Foi desempenhado o CRUD das notas, ajuste visual da validação, tela de agenda e horários e, por fim, continuar avançando na documentação.
- **Sexta Sprint:** Focou-se em realizar o *mockup* das atividades, *home*, notas, ausências, eventos e horários e *back-end* do CRUD das notas.

- **Sétima Sprint:** Efetuou-se o planejamento da nova arquitetura do aplicativo, adotando o padrão MVC para o módulo da *splash screen*, fazer ajustes visuais para a página inicial do aplicativo, adotar padrão MVC para o módulo de cadastro e realizar testes referentes ao novo padrão de desenvolvimento adotado.
- **Oitava Sprint:** Planejou-se que iria-se fazer a criação da validação de e-mail utilizando *regex*, criação de validação de senhas utilizando *regex*, testes relacionados ao no padrão adotado, testes de criação de contas com e-mails já cadastrados, ajustes visuais para página de *login* e adotar padrão MVC para o módulo de *login*.
- **Nona Sprint:** Combinou-se que iria-se fazer a adição do novo padrão MVC para a *homepage*, testar a utilização do *plugin auto\_size\_text*, criar *widgets* compartilhados para reutilizar-se em outros módulos do aplicativo, estudos referente ao *plugin calendar\_table* para utilizar-se no módulo de agenda e por fim testes referentes a implementação da nova arquitetura MVC.
- **Décima Sprint:** Decidiu-se que iria-se fazer a incorporação do novo padrão MVC para o módulo de disciplinas, fazer ajustes visuais no módulo do perfil, criação de tela para mudança de *e-mail*, senha e nome de usuário, criação de logica para mudança de *e-mail*, senha e nome de usuário e por fim uma reestruturação do banco de dados para módulos de ausências e notas.
- **Décima primeira Sprint:** As tarefas decididas foram: adotar padrão MVC para módulo de notas, adotar padrão MVC para módulo de ausências, resolução de *bug* no módulo de perfil ao selecionar foto do usuário e reestruturação do banco de dados para módulos de eventos e horários.
- **Décima segunda Sprint:** Focou-se em adotar padrão MVC para o módulo de horários, adotar padrão MVC para módulo de agenda, implementação do *plugin auto\_size\_text* em todas as telas do aplicativo, testes em dispositivos menores visando a responsividade do aplicativo, ajustando diagramas da documentação.
- **Décima terceira Sprint:** Realizou-se testes em todos os fluxos do aplicativo, melhorias visuais, adequações com *figma* na página inicial, criação de tela de erro genérica, criação de redes sociais do aplicativo para divulgação.

- **Décima quarta *sprint*:** Planejou-se realizar refatoração de código do módulo de disciplina, refatoração de código do módulo de notas, refatoração de código do módulo de ausências, refatoração de código do módulo de horários, refatoração de código do módulo de Agenda, adequações do aplicativo para tentativa de publicação na *Play Store*, publicação do aplicativo na *Play Store*, realizar a conclusão e considerações finais da documentação.

### 3.4.1. *Sprint* 1

No dia 14 de março de 2022 o grupo estava em aula e aproveitou para realizar o planejamento do primeiro entregável (15 dias de desenvolvimento, com prazo para 28 de março de 2022). Nesse dia a equipe iniciou um levantamento de ideias de quais linguagens e tecnologias seriam usadas. De modo geral decidiu-se usar o *Flutter* e o *Visual Studio*. Depois disso iniciou-se o desenvolvimento do *design* do projeto com cada membro dando suas opiniões e pontos que poderiam ser adicionados.

A Tabela 4 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 4 - Planejamento realizado para primeira entrega**

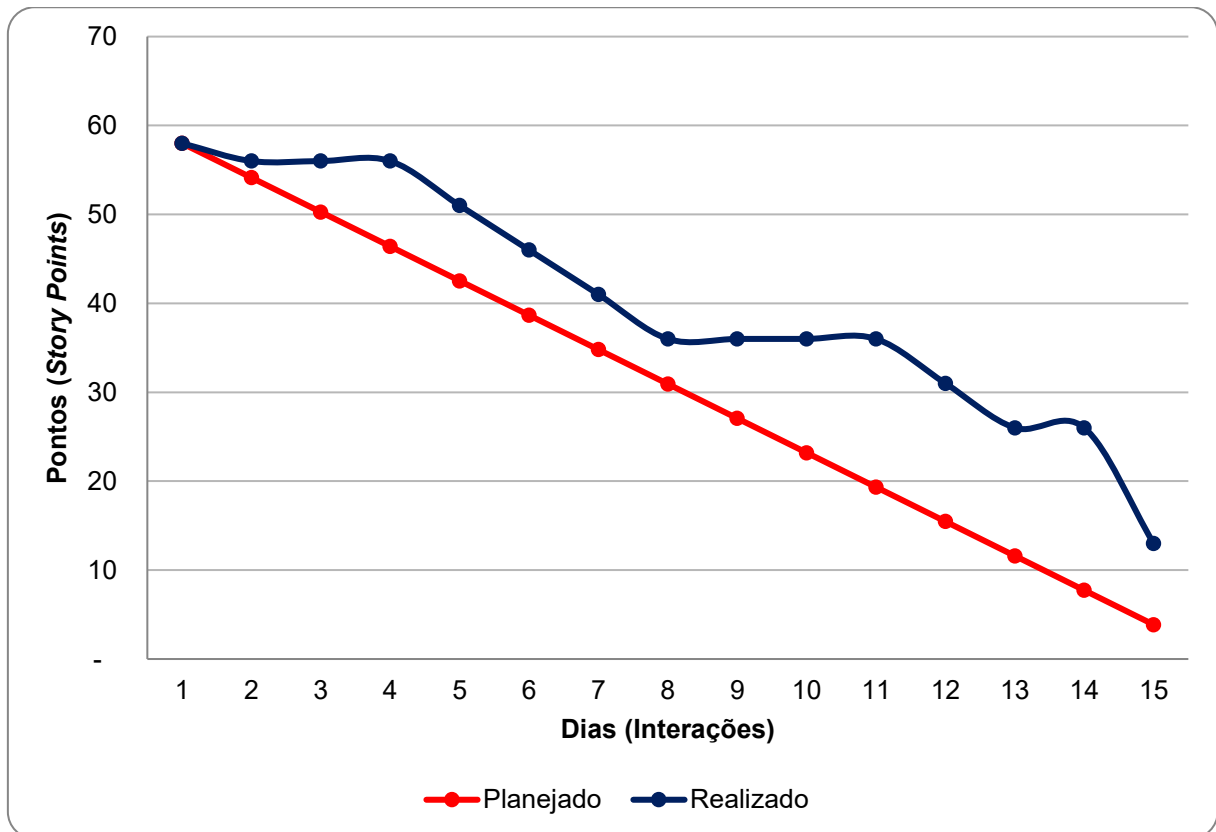
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Protótipo de baixa fidelidade das telas	2h	2
Protótipo de alta fidelidade da tela de <i>login</i> e <i>home</i> ( <i>Figma</i> )	1	5
Protótipo de alta fidelidade do CRUD de disciplinas ( <i>Figma</i> )	1	5
Protótipo de alta fidelidade do CRUD de notas ( <i>Figma</i> )	1	5
Protótipo de alta fidelidade do CRUD de ausências ( <i>Figma</i> )	1	5
Protótipo de alta fidelidade do CRUD de horários ( <i>Figma</i> )	1	5
Protótipo de alta fidelidade do CRUD da agenda ( <i>Figma</i> )	1	5
Tela de <i>login</i> com <i>Google</i> ( <i>Front-end</i> )	7	13
Tela de <i>login</i> com <i>Google</i> ( <i>Back-end</i> )	7	13
<b>Total</b>	<b>20</b>	<b>58</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizada baixas na pontuação conforme as atividades eram concluídas.

A Figura 1 apresenta o gráfico de *Burndown* da entrega 1.

Figura 1 – Gráfico de *Burndown* da entrega 1.



Fonte: Elaborado pelos autores.

Nos dias 26 e 27 de março de 2022 a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Conseguiu-se desenhar quase todas as telas no *Figma*, fazer a escolha das tecnologias do aplicativo e por fim realizar um estudo comparativo dos concorrentes para entender suas funcionalidades;
- O que deu errado: A quebra das tarefas não foi muito bem realizada e houve uma falta significativa de planejamento;
- Ações de melhorias: Melhorar o planejamento e divisão das tarefas, para assim nossa equipe conseguir se organizar e realizar com precisão as tarefas;



### 3.4.2. *Sprint 2*

No dia 28 de março de 2022, parte do grupo se reuniu para realizar o planejamento do segundo entregável (15 dias de desenvolvimento – prazo 11 de abril de 2022). Nesse dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. De modo geral decidiu-se criar o protótipo de alta fidelidade das telas no modo escuro, a interface das telas “*home*” e “*user*” e também tentou-se fazer o *login* com o *Google* funcionar.

A Tabela 5 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

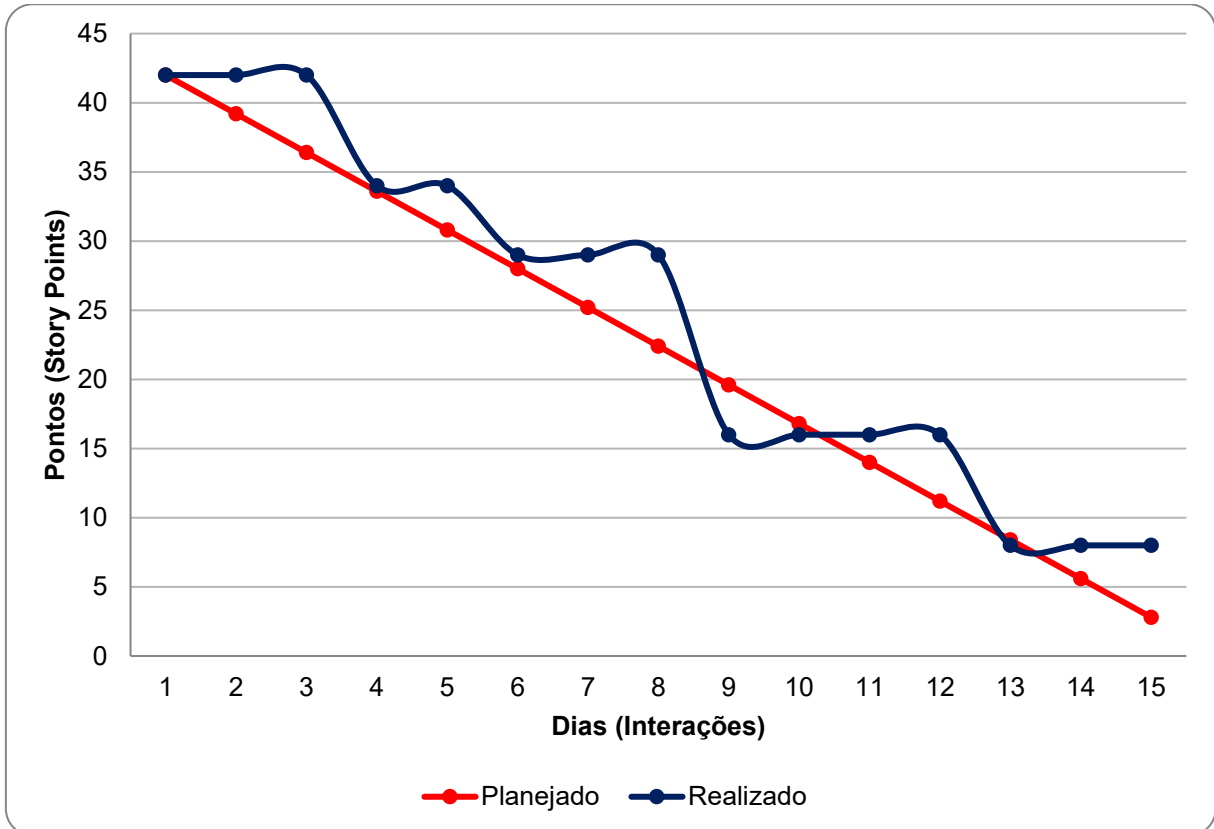
**Tabela 5 – Planejamento realizado para segunda entrega**

<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Protótipo de alta fidelidade de todas as telas modo escuro ( <i>Figma</i> )	7	13
Desenvolvimento da <i>splash screen</i>	1	5
<i>Login</i> com <i>Google</i> funcionando	3	8
Desenvolvimento da interface da <i>home page</i>	3	8
Desenvolvimento da interface da perfil <i>page</i>	3	8
<b>Total</b>	<b>17</b>	<b>42</b>

**Fonte:** Elaborado pelos autores.

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação, conforme as atividades iam sendo concluídas.

A Figura 2 apresenta o gráfico de *Burndown* da entrega 2.

Figura 2 – Gráfico de *Burndown* da entrega 2

Fonte: Elaborado pelos autores.

Nos dias 9 e 10 de abril de 2022, a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião quase todos os membros estavam presentes. Durante a reunião, a equipe tentou concluir o desenvolvimento do *login* com o *Google*, porém ele não funcionou na máquina de todos os integrantes da equipe. Por essa razão, foi considerado como não concluído. Também se realizou uma autoavaliação, procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Conseguiu-se concluir praticamente todas as tarefas planejadas. Foi realizada uma quebra melhor das tarefas e do planejamento, e todos os membros da equipe agora estão com o ambiente de desenvolvimento funcionando;
- O que deu errado: O *login* com o *Google* não funcionou corretamente, pois funcionava apenas para um integrante da equipe, e a distribuição do tempo não foi adequada;

- Ações de melhorias: Melhorar a distribuição do tempo;

### 3.4.3. *Sprint 3*

No dia 11 de abril de 2022, parte do grupo se reuniu para realizar o planejamento do terceiro entregável (dessa vez foram 22 dias de desenvolvimento pois houve uma semana em que não teve aula). Nesse dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Decidiu-se fazer ajustes e melhorias na interface de praticamente todas as telas, protótipo de alta fidelidade das telas “entrar” e “cadastre-se” no *Figma*. Foram desenvolvidas as telas “entrar”, “cadastre-se”, “disciplinas”, *modal* “adicionar disciplina”, “documentação da interface do usuário” e “requisitos e ferramentas”. Por fim, decidiu-se fazer o *login* e-mail e senha padrão funcionar.

A Tabela 6 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 6 – Planejamento realizado para terceira entrega**

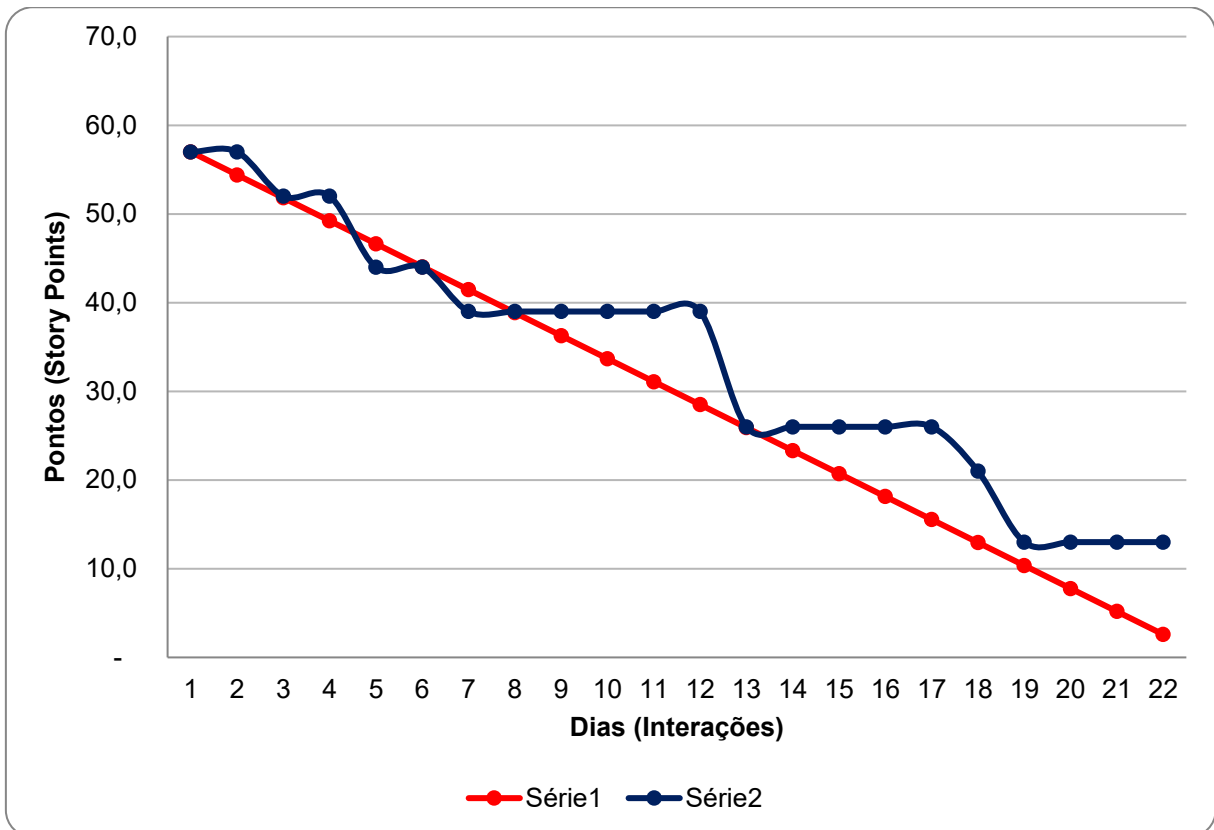
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Ajuste e melhorias em todas as telas	3	8
Protótipo de alta fidelidade das telas entrar e cadastre-se ( <i>Figma</i> )	3	8
Desenvolvimento das telas cadastre-se e entrar	1	5
<i>Login</i> funcionando	7	13
Desenvolvimento da interface da tela disciplinas	1	5
Desenvolvimento da interface do modal adicionar disciplina	1	5
Documentação da interface do usuário e requisitos e ferramentas	7	13
<b>Total</b>	<b>23</b>	<b>57</b>

**Fonte: Elaborado pelos autores**

Durante os 22 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 3 apresenta o gráfico de *Burndown* da entrega 3.

Figura 3 - Gráfico de *Burndown* da entrega 3



Fonte: Elaborado pelos autores.

Nos dias 30 de abril e 1 de maio de 2022, a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião, todos os membros estavam presentes. Durante as reuniões, a equipe tentou concluir o desenvolvimento do *login* e-mail e senha. Porém ele não funcionou corretamente, pois o usuário ficava preso na tela de *login* em um carregamento infinito. Por essa razão, considerou-se como não concluído. Também foi realizada uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Concluiu-se grande parte da *Sprint*, observando-se um bom avanço na documentação da interface de usuário e na de requisitos e ferramentas;
- O que deu errado: O *login e-mail* e senha não estão funcionando corretamente ainda;
- Ações de melhorias: Melhorar a distribuição do tempo;

### 3.4.4. Sprint 4

No dia 02 de maio de 2022, parte do grupo se reuniu para realizar o planejamento do quarto entregável (15 dias de desenvolvimento – prazo 16 de maio de 2022). Nesse dia, a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Decidiu-se tentar fazer o cadastro e *login* padrão funcionar pela segunda vez, validar campos na tela *login* e cadastrar-se, adicionar máscara para o *login*, fazer ajustes de interface, melhorar a documentação, CRUD de disciplinas e notas e por fim um protótipo de alta fidelidade das telas editar perfil e trocar senha.

A Tabela 7 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 7 – Planejamento realizado para quarta entrega**

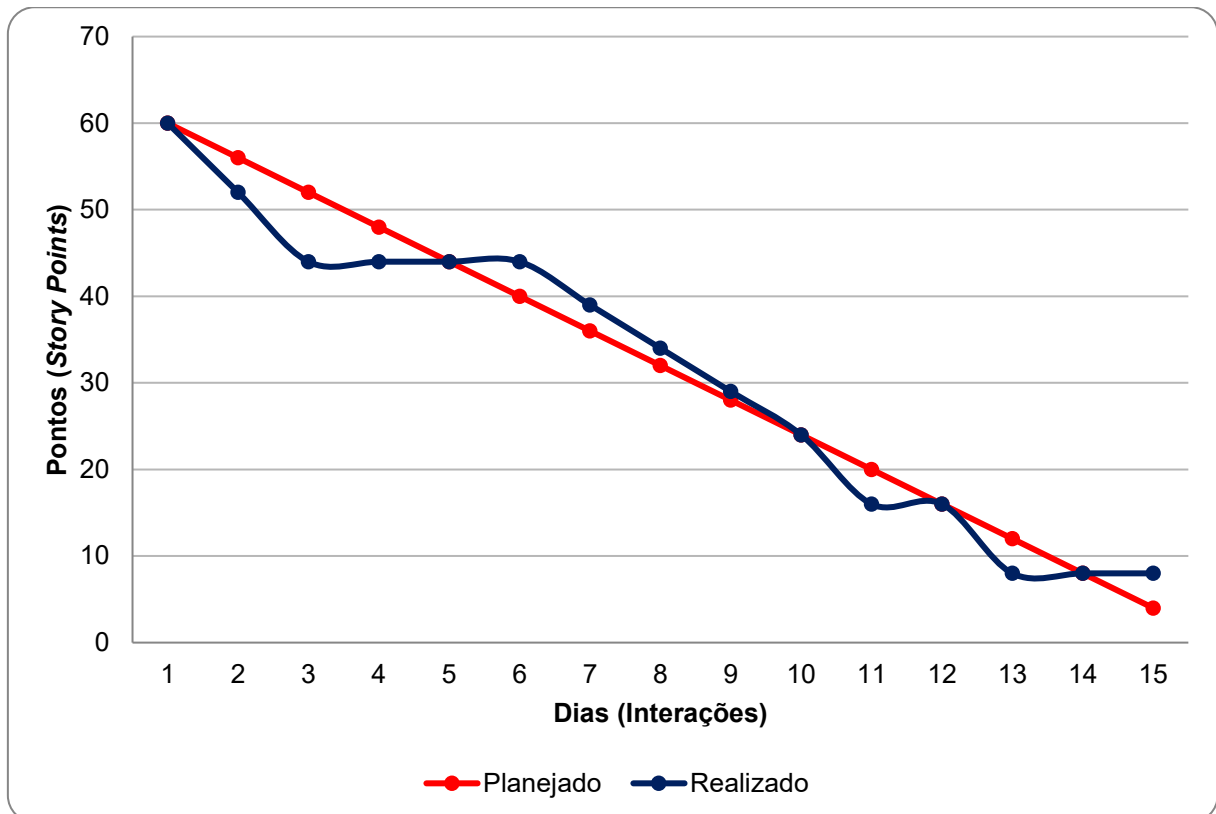
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
<i>Login</i> funcionando	3	8
Cadastrar-se funcionando	3	8
Adicionar máscara para o <i>login</i>	1	5
Validar campos na tela <i>login</i> e cadastrar-se	1	5
Ajustes de interface	1	5
Documentação	3	8
CRUD Disciplinas	3	8
CRUD Notas	3	8
Protótipo de alta fidelidade das telas editar perfil e trocar senha	1	5
<b>Total</b>	<b>19</b>	<b>60</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades iam sendo concluídas.

A Figura 4 apresenta o gráfico de *Burndown* da entrega 4.

Figura 4 - Gráfico de *Burndown* da entrega 4



Fonte: Elaborado pelos autores

No dia 14 de maio de 2022, a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião, apenas dois membros estavam presentes. Durante a reunião, a equipe tentou juntar as alterações de cada *branch*, porém ocorreram conflitos, que provocaram a perda de algumas partes do código e tentou-se fazer o CRUD das notas, porém acabou não dando tempo. Também se realizou uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Cadastro e *login* estão finalmente funcionando, e também conseguiu-se fazer o CRUD das disciplinas;
- O que deu errado: Merge das *branches* no *Git*, cores da validação e o CRUD das notas;
- Ações de melhorias: Melhor divisão do tempo ao longo da semana;

### 3.4.5. *Sprint 5*

No dia 16 de maio de 2022 todo grupo se reuniu para realizar o planejamento do quinto entregável (15 dias de desenvolvimento – prazo 30 de maio de 2022). Nesse dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Decidiu-se realizar CRUD das notas, ajuste visual da validação, tela de agenda e horários e por fim continuar avançando na documentação.

A Tabela 8 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 8 – Planejamento realizado para quinta entrega**

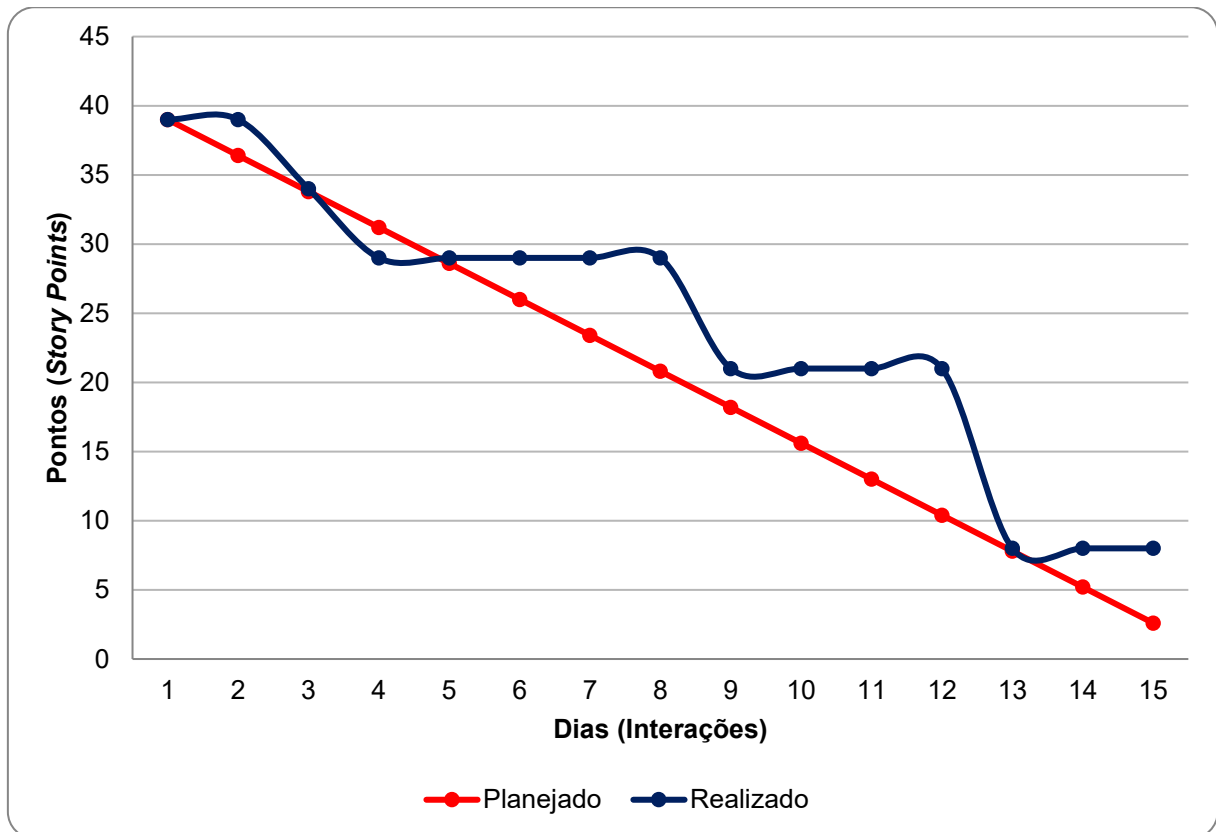
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
CRUD das notas	3	8
Ajuste visual da validação	7	13
Tela agenda	3	8
Tela horários	1	5
Documentação	1	5
<b>Total</b>	<b>15</b>	<b>39</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 5 apresenta o gráfico de *Burndown* da entrega 5.

Figura 5 – Gráfico de *Burndown* da entrega 5



Fonte: Elaborado pelos autores.

No dia 29 de maio de 2022, a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe tentou finalizar o CRUD das notas, porém alguns erros aconteceram e, infelizmente, acabou não dando tempo. A equipe também realizou uma autoavaliação, procurando detectar os principais pontos de acertos e erros, e juntos, propôs ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Ajuste da validação e as telas de agenda e horários;
- O que deu errado: CRUD das notas;
- Ações de melhorias: Focar mais nos momentos de desenvolvimento.

### 3.4.6. *Sprint 6*

No dia 30 de maio de 2022 todo grupo se reuniu para realizar o planejamento do sexto entregável, dessa vez com um tempo um pouco maior por conta de



imprevistos no dia que deveria ser feita a apresentação (21 dias de desenvolvimento – prazo 20 de junho de 2022). Nesse dia, a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Decidiu-se realizar o *mockup* das atividades, home, notas, ausências, eventos e horários e *back-end* do CRUD das notas.

A Tabela 9 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 9 – Planejamento realizado para sexta entrega**

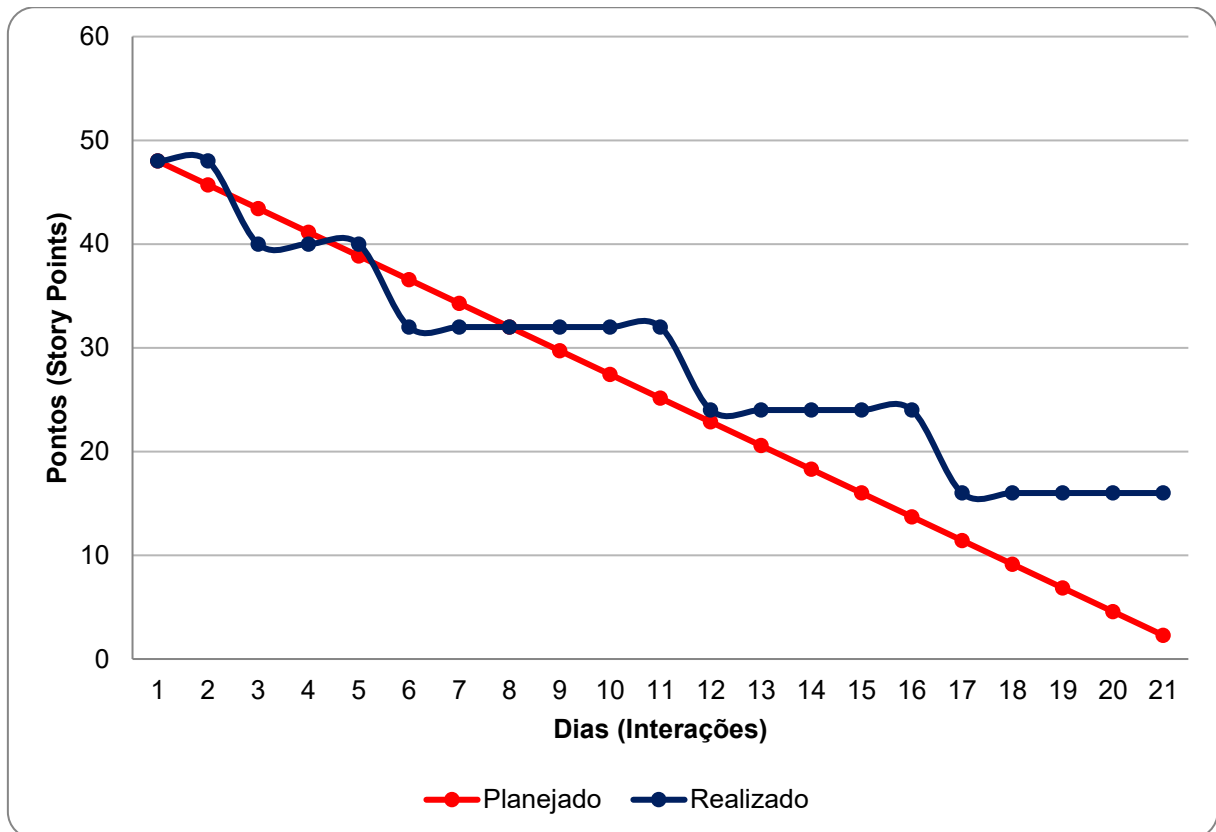
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
<i>Mockup</i> das atividades da home	3	8
<i>Mockup</i> de notas	3	8
<i>Mockup</i> de ausências	3	8
<i>Mockup</i> de eventos	3	8
<i>Mockup</i> de horários	3	8
<i>Back-end</i> do CRUD das notas	3	8
<b>Total</b>	<b>18</b>	<b>48</b>

**Fonte: Elaborado pelos autores.**

Durante os 21 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 6 apresenta o gráfico de *Burndown* da entrega 6.

Figura 6 - Gráfico de *Burndown* da entrega 6



Fonte: Elaborado pelos autores.

No dia 19 de julho de 2022 a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe chegou a pesquisar a fundo sobre como coletar os dados de uma disciplina para assim colocar uma nota nela porém não foi possível, contudo, os *mockups* ficaram praticamente da forma que se queria, também se fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Os *mockups* de dados da parte visual ficaram prontos;
- O que deu errado: CRUD das notas novamente;
- Ações de melhorias: Todos os membros focarem na parte do desenvolvimento, pois agora boa parte da documentação está pronta;

### 3.4.7. *Sprint 7*

Após o termino das férias a equipe voltou-se a fazer as sprints normalmente. No dia 03 de agosto de 2022 todo grupo se reuniu para realizar o planejamento do sétimo entregável, (15 dias de desenvolvimento – prazo 17 de agosto de 2022). Nesse dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Decidiu-se planejar a nova arquitetura do aplicativo, começar adotando o padrão MVC para o módulo da *splash screen*, fazer ajustes visuais para a página inicial do aplicativo, adotar padrão MVC para o módulo de cadastro e realizar testes referentes ao novo padrão de desenvolvimento adotado.

A Tabela 10 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 10 – Planejamento realizado para sétima entrega**

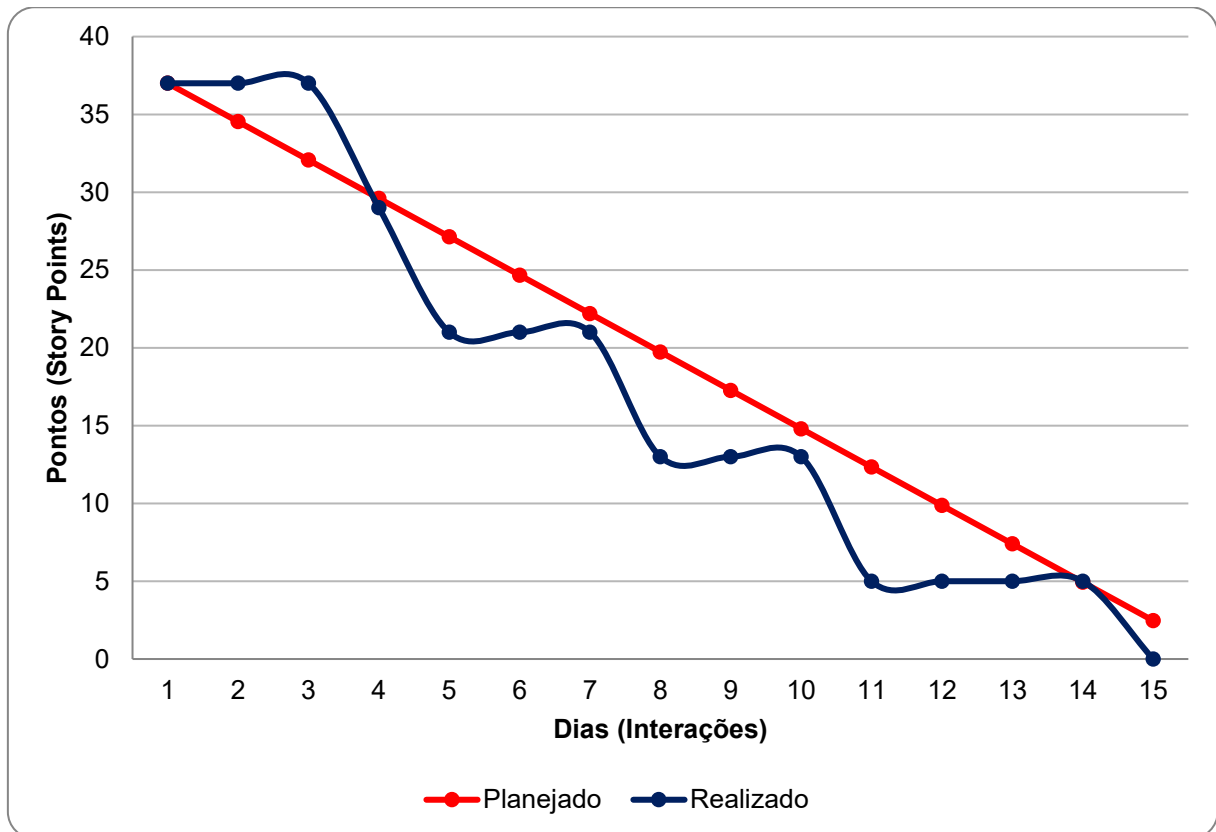
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Planejamento da nova arquitetura do projeto	3	8
Adotar padrão MVC para o módulo da <i>splash screen</i>	3	8
Ajustes visuais para a página inicial do aplicativo	3	8
Adotar padrão MVC para o módulo de cadastro	3	8
Testes referentes ao novo padrão de desenvolvimento adotado	1	5
<b>Total</b>	<b>13</b>	<b>37</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 7 apresenta o gráfico de *Burndown* da entrega 7.

Figura 7 - Gráfico de *Burndown* da entrega 7



Fonte: Elaborado pelos autores.

No dia 17 de agosto de 2022 a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez alguns testes referente ao novo padrão de desenvolvimento, também foi feita uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: O planejamento da nova arquitetura do projeto fluiu bem e definimos o padrão MVC como o novo padrão de desenvolvimento do projeto;
- O que deu errado: Nada;
- Ações de melhorias: Nada.

### 3.4.8. Sprint 8

No dia 17 de agosto de 2022 todo grupo se reuniu para realizar o planejamento do oitavo entregável, (15 dias de desenvolvimento – prazo 31 de agosto de 2022). Nesse dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Planejou-se que iria-se fazer a criação da validação de *e-mail* utilizando *regex*, criação de validação de senhas utilizando *regex*, testes relacionados ao no padrão adotado, testes de criação de contas com *e-mails* já cadastrados, ajustes visuais para página de login e adotar padrão MVC para o módulo de *login*.

A Tabela 11 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

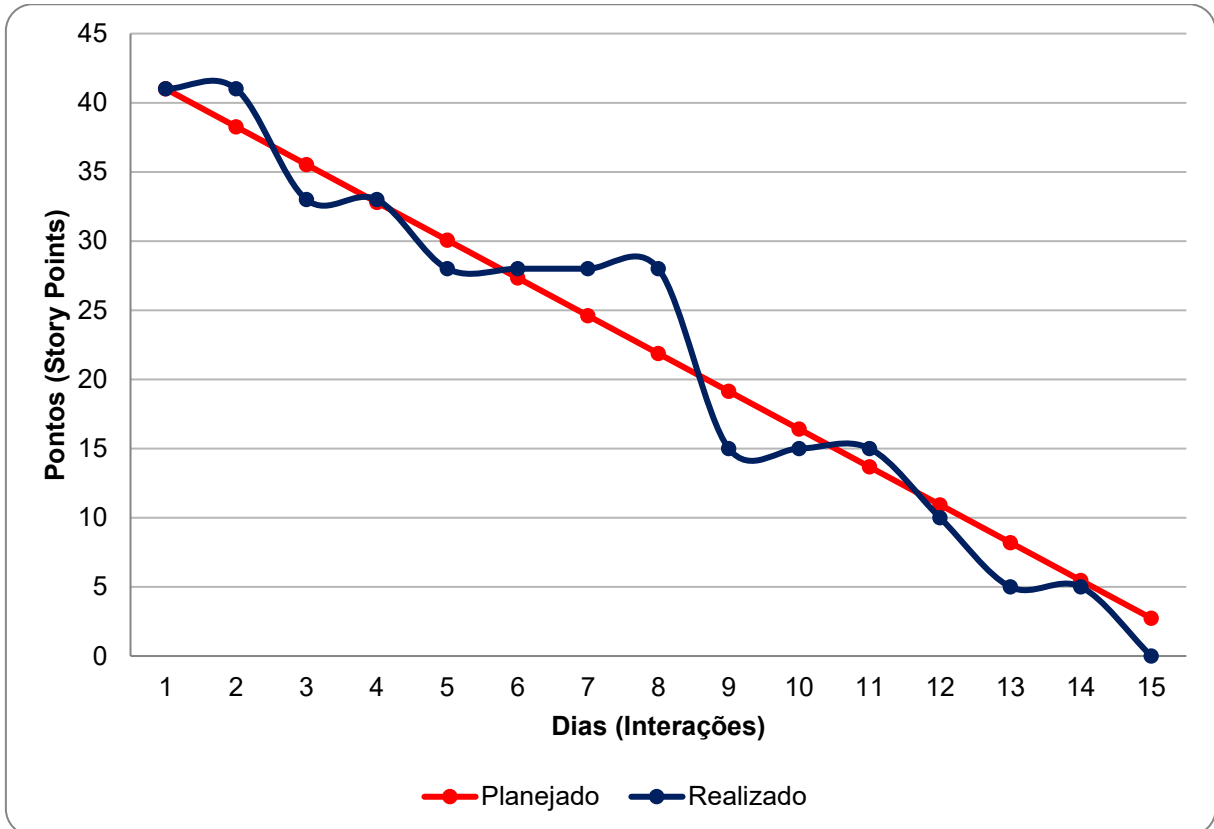
**Tabela 11 - Planejamento realizado para oitava entrega**

<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Criação de validação de <i>e-mail</i> utilizando <i>Regex</i>	1	5
Criação de validação de senhas utilizando <i>Regex</i>	1	5
Testes relacionados ao no padrão adotado	3	8
Testes de criação de contas com <i>e-mails</i> já cadastrados	1	5
Ajustes visuais para página de <i>login</i>	3	5
Adotar padrão MVC para o módulo de <i>login</i>	5	13
<b>Total</b>	<b>14</b>	<b>41</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 8 apresenta o gráfico de *Burndown* da entrega 8.

Figura 8 – Gráfico de *Burndown* da entrega 8

Fonte: Elaborado pelos autores .

No dia 30 de agosto de 2022 a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião a equipe fez alguns testes referente ao novo padrão de desenvolvimento, também foi feita uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Novo padrão MVC no módulo de *login* agiu como esperado de acordo com os testes feitos. Testes de criação de contas com *e-mail* já existentes também funcionaram como o esperado o *firebase* barrou a criação de conta com um *e-mail* já existente;
- O que deu errado: Apesar de termos conseguido entregar o funcionamento das validações de campos utilizando o *Regex* utilizamos mais tempo que o esperado para criar as validações, não levamos em conta o tempo para estudo do *Regex*;

- Ações de melhorias: Pensar melhor no tempo gasto para estudar uma tecnologia.

### 3.4.9. *Sprint 9*

No dia 31 de agosto de 2022 o grupo se reuniu para realizar o planejamento do nono entregável, (15 dias de desenvolvimento – prazo 14 de setembro de 2022). Nesse dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Combinou-se que iria-se fazer a adição do novo padrão MVC para a *homepage*, testar a utilização do *plugin auto\_size\_text*, criar *widgets* compartilhados para reutilizar-se em outros módulos do aplicativo, estudos referente ao *plugin calendar\_table* para utilizar-se no módulo de agenda e por fim testes referentes a implementação da nova arquitetura MVC.

A Tabela 12 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 12 - Planejamento realizado para nona entrega**

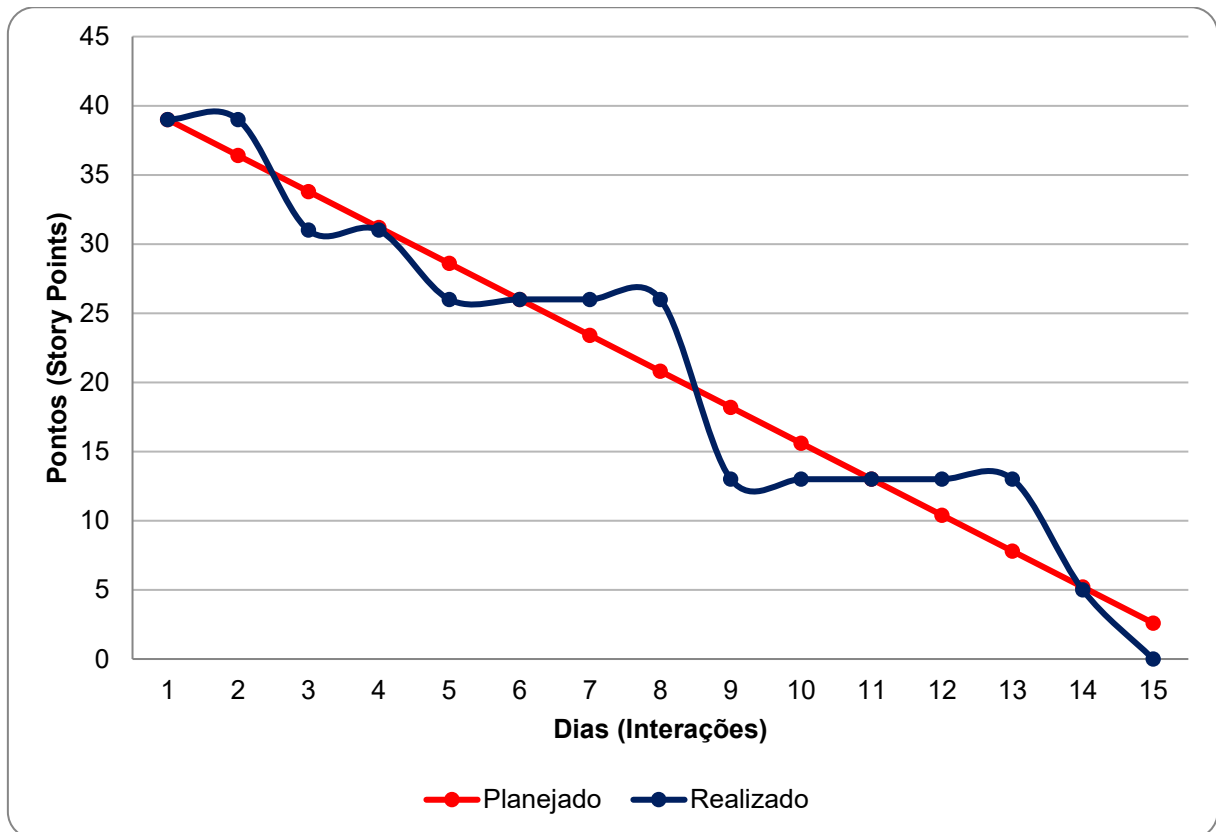
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Adotar novo padrão MVC para a <i>homepage</i>	3	8
Testar a utilização do <i>plugin auto_size_text</i> para evitar-se <i>bug</i> de <i>overflow</i>	1	5
Criar <i>widgets</i> compartilhados para reutilizar-se em outros módulos do aplicativo	5	13
Estudos referente ao <i>plugin calendar_table</i> para utilizarmos no módulo de agenda	3	8
Testes referentes a implementação da nova arquitetura MVC	1	5
<b>Total</b>	<b>14</b>	<b>41</b>

**Fonte: Elaborado pelos autores .**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 9 apresenta o gráfico de *Burndown* da entrega 9.

Figura 9 – Gráfico de *Burndown* da entrega 9.



Fonte: Elaborado pelos autores .

No dia 13 de setembro de 2022 a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião foi feita uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: A nova implementação do padrão MVC foi um sucesso ocorreu tudo como o esperado. A criação de *widgets* universais para poder-se utilizar em outras partes do código e ter-se uma facilidade na manutenção;
- O que deu errado: Nada;
- Ações de melhorias: Nada.



### 3.4.10. *Sprint 10*

No dia 14 de setembro de 2022 o grupo se reuniu para realizar o planejamento do nono entregável, (15 dias de desenvolvimento – prazo 28 de setembro de 2022). Nesse dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo da *sprint*. Decidiu-se que iria-se fazer a incorporação do novo padrão MVC para o módulo de disciplinas, fazer ajustes visuais no módulo do perfil, criação de tela para mudança de *e-mail*, senha e nome de usuário, criação de lógica para mudança de *e-mail*, senha e nome de usuário e por fim uma reestruturação do banco de dados para módulos de ausências e notas.

A Tabela 13 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 13 - Planejamento realizado para décima entrega**

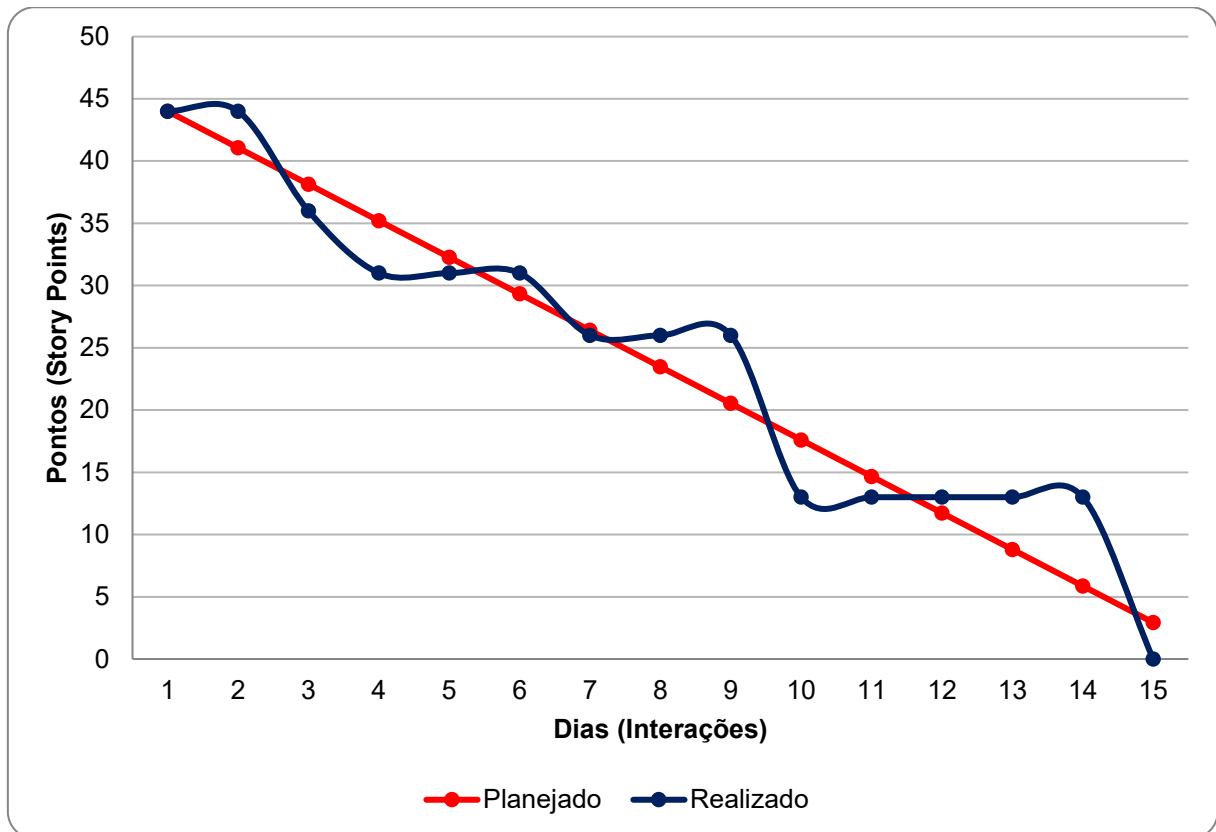
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Adotar novo padrão MVC para o módulo de disciplinas	3	8
Ajustes visuais no módulo do perfil	1	5
Criação de tela para mudança de <i>e-mail</i> , senha e nome de usuário	1	5
Criação de lógica para mudança de <i>e-mail</i> , senha e nome de usuário	5	13
Reestruturação do banco de dados para módulos de ausências e notas	5	13
<b>Total</b>	<b>15</b>	<b>44</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 10 apresenta o gráfico de *Burndown* da entrega 10.

Figura 10 – Gráfico de *Burndown* da entrega 10



Fonte: Elaborado pelos autores.

No dia 27 de setembro de 2022 a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião foi feita uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Todas as tarefas deram certo;
- O que deu errado: Nada;
- Ações de melhorias: Nada.

### 3.4.11. *Sprint 11*

No dia 28 de setembro de 2022 o grupo se reuniu para realizar o planejamento do nono entregável, (15 dias de desenvolvimento – prazo 12 de outubro de 2022). Neste dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo da sprint. As tarefas decididas foram: adotar padrão MVC para módulo de notas,

adotar padrão MVC para módulo de ausências, resolução de *bug* no módulo de perfil ao selecionar foto do usuário e reestruturação do banco de dados para módulos de eventos e horários.

A Tabela 14 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 14 - Planejamento realizado para décima primeira entrega**

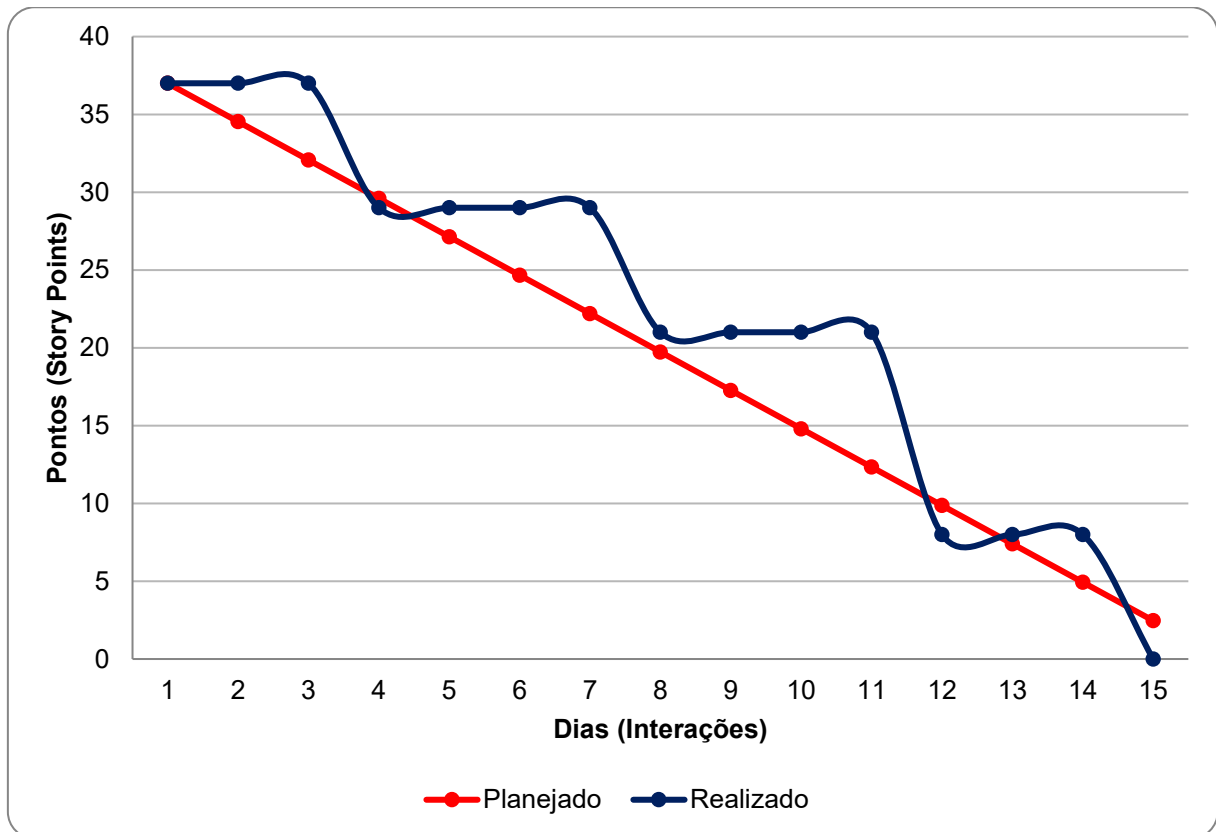
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Adotar padrão MVC para módulo de notas	3	8
Adotar padrão MVC para módulo de ausências	3	8
Resolução de <i>bug</i> no módulo de perfil ao selecionar foto do usuário	3	8
Reestruturação do banco de dados para módulos de eventos e horários	5	13
<b>Total</b>	<b>14</b>	<b>37</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 11 apresenta o gráfico de *Burndown* da entrega 11.

Figura 11 – Gráfico de *Burndown* da entrega 11



Fonte: Elaborado pelos autores .

No dia 11 de setembro de 2022 a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião foi feita uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Todas as tarefas deram certo;
- O que deu errado: Nada;
- Ações de melhorias: Nada.

### 3.4.12. *Sprint 12*

No dia de 12 de outubro de 2022, todo grupo se reuniu para realizar o planejamento do décimo segundo entregável (15 dias de desenvolvimento – prazo 26 de outubro de 2022). Nesse dia, a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Decidiu-se adotar padrão MVC para o módulo de

horários, adotar padrão MVC para módulo de agenda, implementação do *plugin auto\_size\_text* em todas as telas do aplicativo, testes em dispositivos menores visando a responsividade do aplicativo, ajustando diagramas da documentação.

A Tabela 15 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 15 - Planejamento realizado para décima segunda entrega**

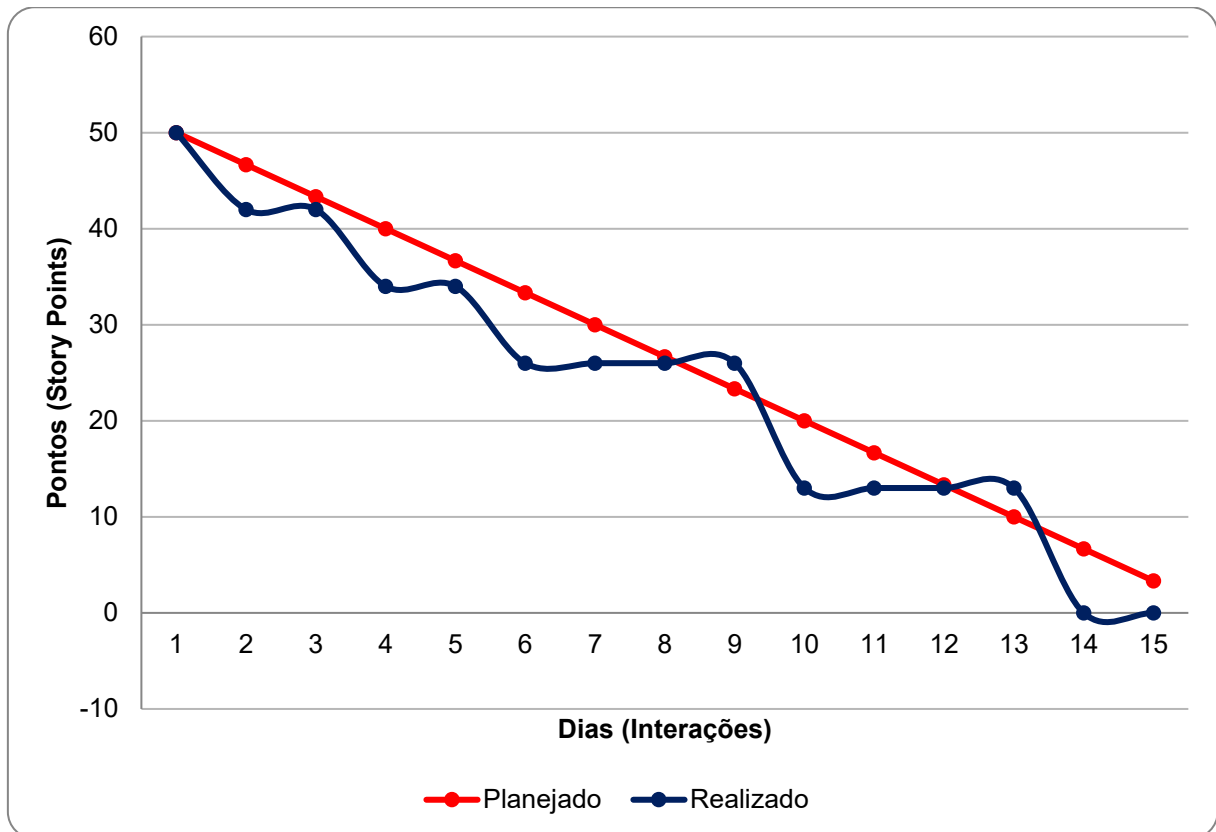
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Adotar padrão MVC para módulo de horários	3	8
Adotar padrão MVC para módulo de agenda	3	8
Implementação do <i>plugin auto_size_text</i> em todas as telas do aplicativo	5	13
Testes em dispositivos menores visando a responsividade do aplicativo	3	8
Ajustando diagramas da documentação	5	13
<b>Total</b>	<b>19</b>	<b>50</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 12 apresenta o gráfico de *Burndown* da entrega 12.

Figura 12 – Gráfico de *Burndown* da entrega 12.



Fonte: Elaborado pelos autores.

Na data de 26 de outubro de 2022, a equipe se reuniu para fazer uma reunião para revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe revisou todos entregáveis que foram planejados para essa *sprint*, realizamos uma análise dos pontos positivos e negativos que ocorreram durante este período, foi possível concluir que o time teve um ótimo desempenho e realizou todas as atividades de forma rápida e eficaz. Entretanto, alguns pontos de atenção foram observados, como: melhorias no processo de desenvolvimento, padronização no versionamento, utilização de ferramentas de análise de códigos.

- O que deu certo: Implementação do padrão de arquitetura MVC;
- O que deu errado: Dificuldade na padronização de versionamento;
- Ações de melhorias: Focar em técnicas para facilitar o trabalho em equipe no desenvolvimento dos sistemas.

### 3.4.13. *Sprint 13*

No dia 26 de outubro de 2022 todo grupo se reuniu para realizar o planejamento do décimo terceiro entregável (15 dias de desenvolvimento – prazo 09 de novembro de 2022). Nesse dia, a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Decidiu-se realizar testes em todos os fluxos do aplicativo, melhorias visuais, adequações com *figma* na página inicial, criação de tela de erro genérica, criação de redes sociais do aplicativo para divulgação.

A Tabela 16 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

**Tabela 16 - Planejamento realizado para décima terceira entrega**

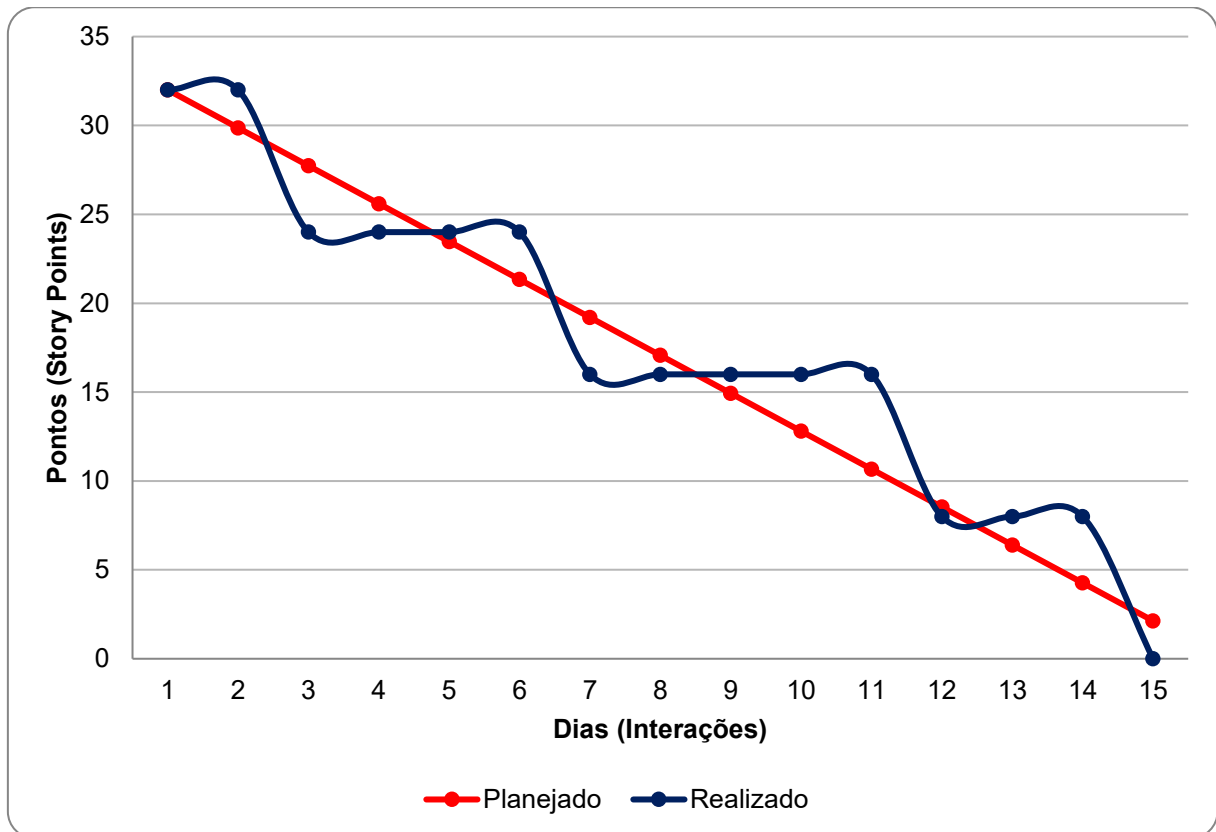
<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Testes em todos os fluxos do aplicativo	3	8
Melhorias visuais e adequações com figma na página inicial	3	8
Criação de tela de erro genérica	3	8
Criação de redes sociais do aplicativo para divulgação	3	8
<b>Total</b>	<b>12</b>	<b>32</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 13 apresenta o gráfico de *Burndown* da entrega 13.

Figura 13 - Gráfico de *Burndown* da entrega 13



Fonte: Elaborado pelos autores.

No dia 09 de novembro de 2022 a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, foi discutido diversos pontos sobre o resultado dos testes que foram feitos em todos os fluxos do aplicativo e discutimos diversas melhorias que poderiam ser feitas na próxima sprint para a finalização do projeto.

- O que deu certo: Testes em todos os fluxos no aplicativo
- O que deu errado: Vários *bugs* novos identificados que ocuparam tempo que seria realizado em outras demandas do projeto;
- Ações de melhorias: Focar na solução mais rápida e prática para os problemas de desenvolvimento.

#### 3.4.14. *Sprint 14*

No dia 09 de novembro de 2022, todo grupo se reuniu para realizar o planejamento do décimo quarto entregável (15 dias de desenvolvimento – prazo 23



de novembro de 2022). Nesse dia a equipe decidiu quais seriam os tópicos a serem desenvolvidos ao longo dos dias. Decidiu-se realizar refatoração de código do módulo de disciplina, refatoração de código do módulo de notas, refatoração de código do módulo de ausências, refatoração de código do módulo de horários, refatoração de código do módulo de Agenda, adequações do aplicativo para tentativa de publicação na *Play Store*, publicação do aplicativo na *Play Store*, realizar a conclusão e considerações finais da documentação.

A Tabela 17 apresenta detalhadamente as atividades, seu tempo de realização em dias e sua respectiva pontuação.

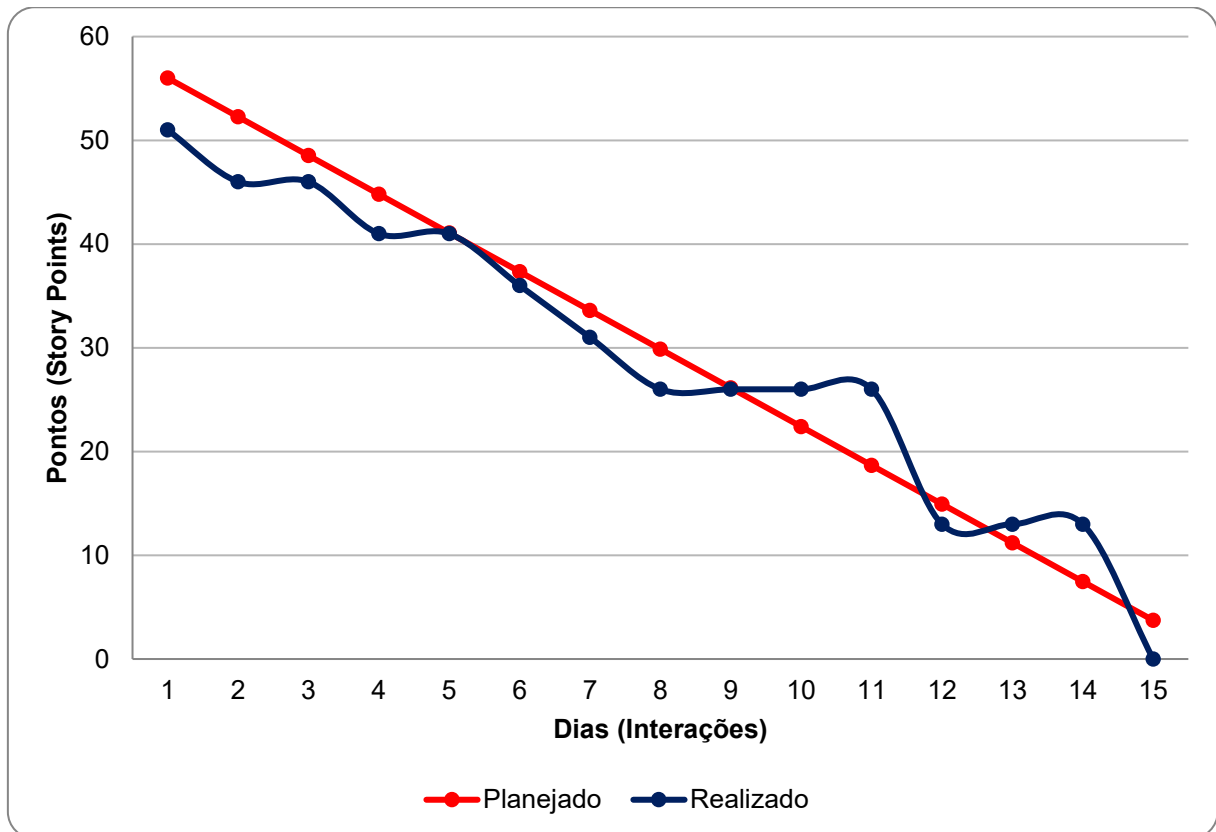
**Tabela 17 - Planejamento realizado para décima quarta entrega**

<b>Atividade</b>	<b>Tempo</b>	<b>Pontos</b>
Refatoração de código do módulo de disciplina	1	5
Refatoração de código do módulo de Notas	1	5
Refatoração de código do módulo de Ausências	1	5
Refatoração de código do módulo de horários	1	5
Refatoração de código do módulo de Agenda	1	5
Adequações do aplicativo para tentativa de publicação na <i>Play Store</i>	5	13
Publicação do aplicativo na <i>Play Store</i>	1	5
Fazendo a conclusão e considerações finais na documentação	5	13
<b>Total</b>	<b>16</b>	<b>56</b>

**Fonte: Elaborado pelos autores.**

Durante os 15 dias de desenvolvimento, a equipe realizou baixas na pontuação conforme as atividades eram concluídas.

A Figura 14 apresenta o gráfico de *Burndown* da entrega 14.

Figura 14 – Gráfico de *Burndown* da entrega 14

Fonte: Elaborado pelos autores.

Na data de 23 de novembro de 2022, a equipe se reuniu para fazer uma reunião de revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe finalizou os procedimentos para publicação do aplicativo, analisamos os pontos que poderiam ser melhorados na documentação do projeto, finalizamos a conclusão e as considerações finais.

- O que deu certo: Refatoração de diversos módulos do projeto, publicação e conclusão da documentação;
- O que deu errado: Nada.
- Ações de melhorias: Nada.

### 3.5. Desenvolvimento do protótipo de alta fidelidade

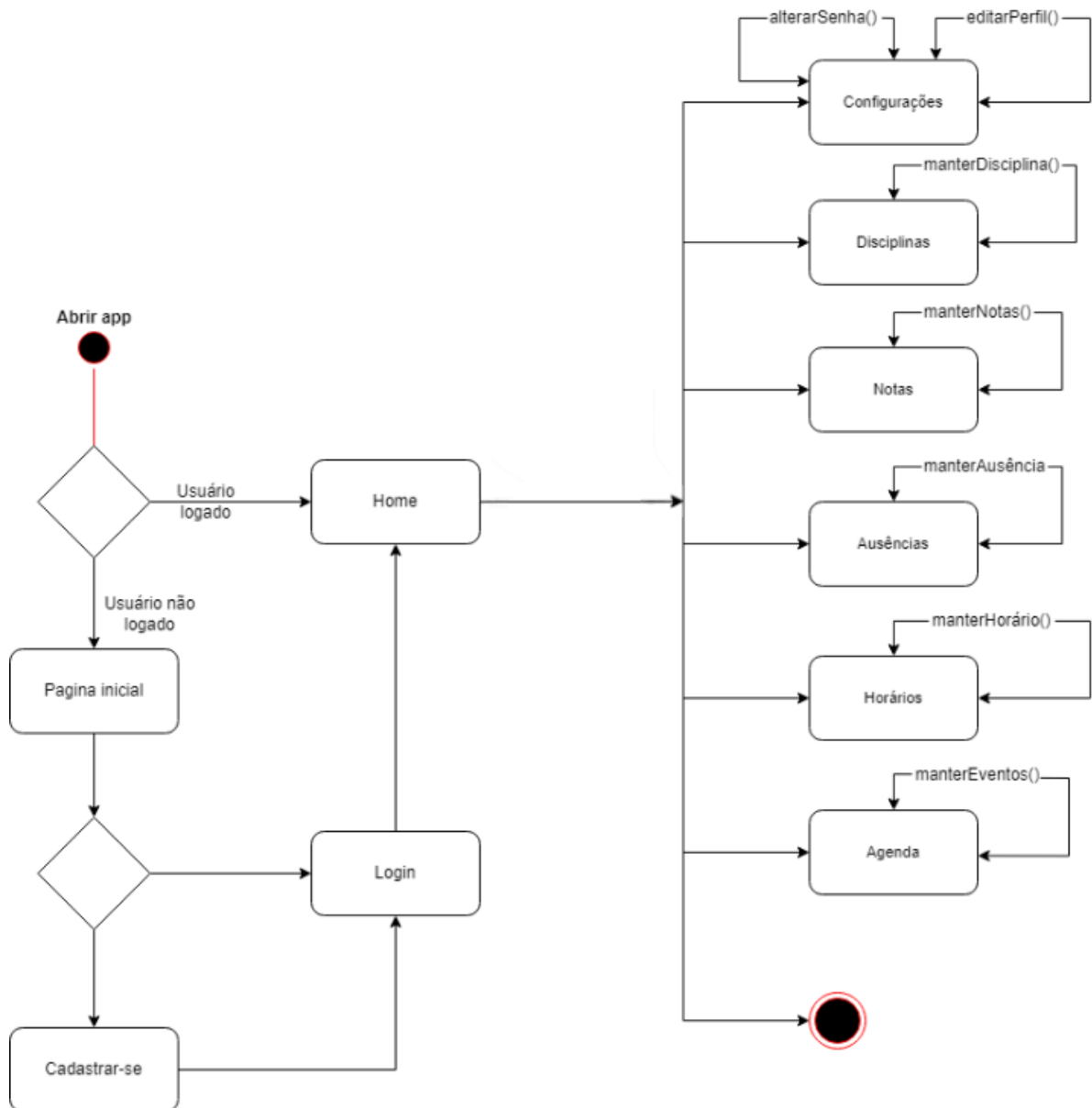
A necessidade da construção de uma interface amigável ao usuário é fundamental em um sistema. A interface faz parte do sistema computacional e determina como as pessoas operam e controlam o sistema. Quando uma interface é

bem projetada, é compreensível, agradável e controlável. Neste contexto, fez-se um protótipo de alta fidelidade utilizando-se do *software Figma*. Os protótipos têm como objetivo apresentar todas as telas do aplicativo, da Figura 16 até a Figura 48

Inicialmente vamos apresentar o diagrama de estados que é um mapa das telas em engenharia de *software* é uma representação do estado ou situação em que um objeto pode se encontrar no decorrer da execução de processos de um sistema.

A Figura 15 apresenta o Diagrama de estados do aplicativo.

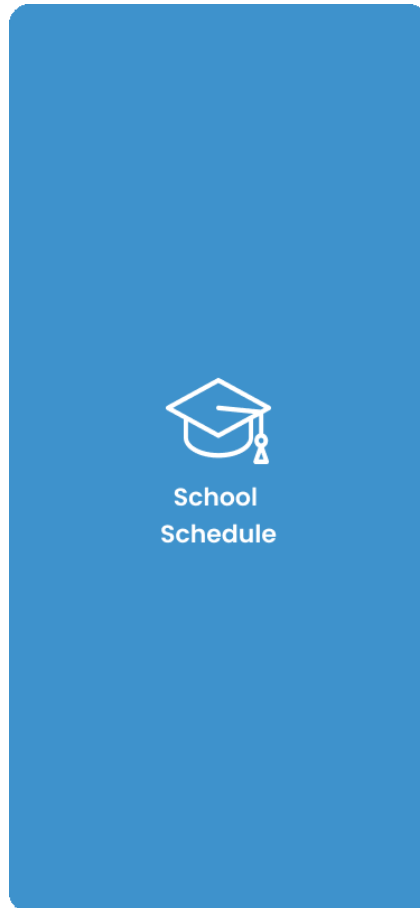
**Figura 15 – Diagrama de estados (mapa das telas).**



**Fonte: Elaborado pelos autores.**

A Figura 16 apresenta a *splash screen* do aplicativo.

**Figura 16 - Captura da *Splash Screen***

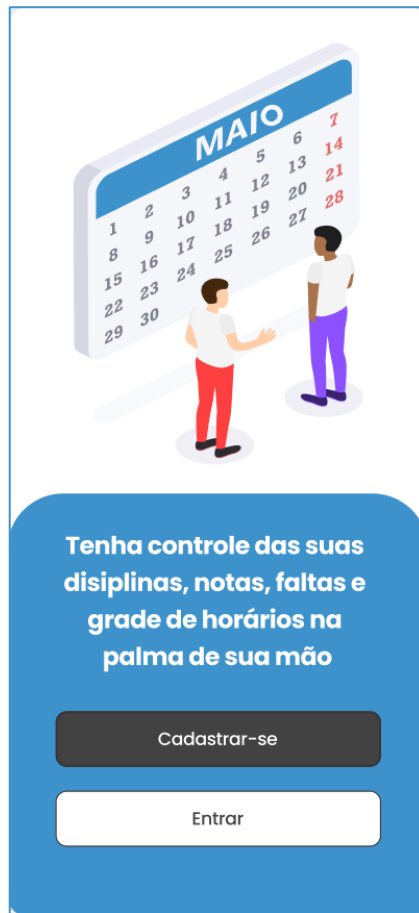


**Fonte: Elaborado pelos autores**

A *splash screen* na Figura 16, é composta por:

- **Ícone:** O ícone do aplicativo.
- **Texto:** Um texto com o nome do aplicativo.

A Figura 17 apresenta a tela inicial do aplicativo, onde o usuário pode escolher entre fazer um cadastro ou entrar com sua conta, caso ele esteja logado essa tela não aparecerá.

**Figura 17 – Captura da Tela Inicial**

Fonte: Elaborado pelos autores

A tela inicial apresentada na Figura 17, é composta por:

- **Botão Cadastrar-se:** Para ir até a tela de cadastro do aplicativo.
- **Botão Entrar:** Para conseguir ir até a tela entrar do aplicativo.

A Figura 18 apresenta a tela de cadastro, no qual o usuário preenche os campos para criar uma conta, sendo que todos os campos são obrigatórios.

Figura 18 – Captura da tela Cadastro



**Cadastre-se**

Nome de usuário  
Marcos Rodrigues

Email  
usuario@gmail.com

Senha  
●●●●●●

Confirme a senha  
●●●●●●

**Cadastrar-se**

Já tenho um cadastro. [Entrar](#)

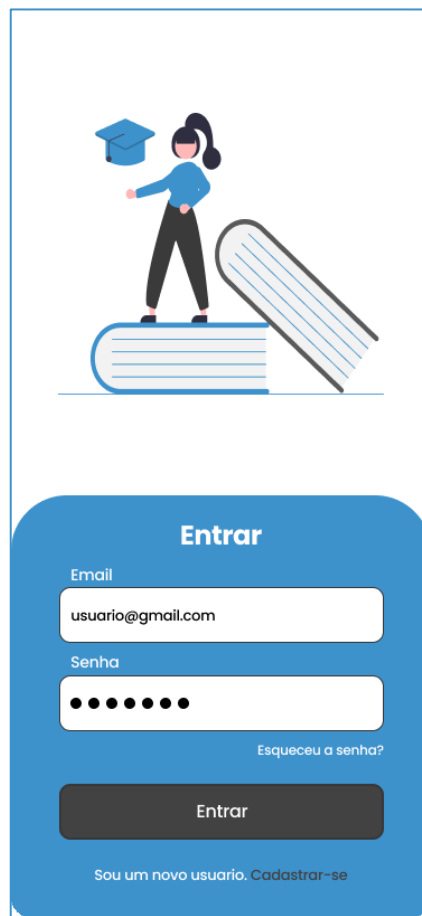
Fonte: Elaborado pelos autores

A tela de cadastro apresentada na Figura 18, é composta por:

- **Campo nome do usuário:** Para colocar o nome do usuário.
- **Campo E-mail:** Para colocar o e-mail do usuário.
- **Campo senha:** Para colocar a senha do usuário.
- **Campo confirme senha:** Para colocar novamente a senha do campo anterior.
- **Botão Cadastrar:** Ao clicar nesse botão, o aplicativo salva as informações, assim o usuário consegue logar no aplicativo.
- **Botão Entrar:** Caso o usuário já tenha uma conta ele pode clicar no botão entrar e ir direto para a tela de *login*.

A Figura 19 apresenta a tela de *login*, na qual o usuário coloca suas informações já cadastradas para entrar no aplicativo.

Figura 19 - Captura da Tela de *Login*



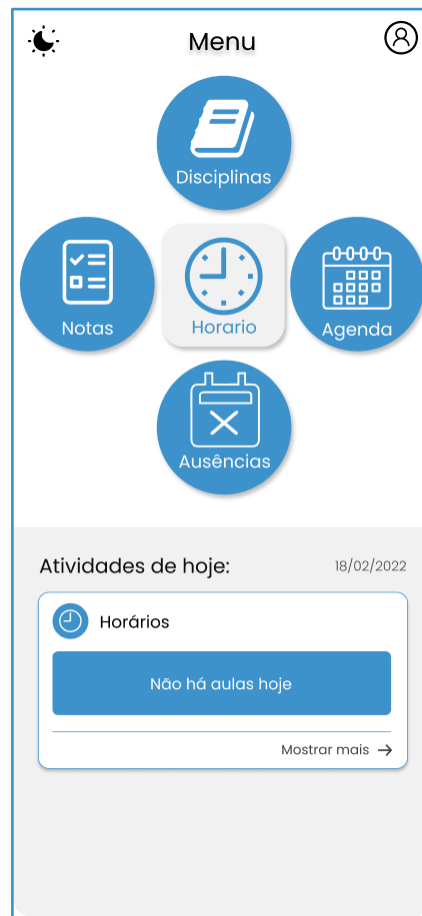
Fonte: Elaborado pelos autores

A tela “*Home*” apresentada na Figura 19, é composta por:

- **Campo *E-mail*:** Para colocar o *e-mail*.
- **Campo senha:** Para colocar a senha.
- **Botão esqueceu a senha:** Para direcionar à tela de recuperação da senha.
- **Botão entrar:** Para direcionar a tela *home*.
- **Botão cadastrar-se:** Caso o usuário ainda não tenha uma conta ele pode clicar no botão cadastrar e ir direto para a tela cadastrar-se.

A Figura 20 apresenta a tela “*Home*”, na qual se consegue visualizar vários botões que vão para diversas telas, e, rolando para baixo, é possível ver os horários e as atividades do dia atual.

**Figura 20 - Captura da Tela *Home*.**



**Fonte: Elaborado pelos autores**

A tela “*Home*”, apresentada na Figura 20 é composta por:

- **Botão trocar de tema:** Para alterar o tema o aplicativo entre modo claro e escuro.
- **Botão configurações:** Para direcionar a tela que contém as informações do usuário e do aplicativo.
- **Botão disciplinas:** Para direcionar à tela das disciplinas.
- **Botão notas:** Para direcionar à tela das notas.
- **Botão horário:** Para direcionar à tela dos horários.
- **Botão agenda:** Para direcionar à tela agenda.
- **Botão ausências:** Para direcionar à tela das ausências.
- **Botão mostrar mais:** Ao clicar nesse botão, direciona-se para a tela que mostra mais informações sobre aquela atividade.



A Figura 21 apresenta a tela de configurações, no qual o usuário consegue sair de sua conta, fazer alterações no seu perfil e ver algumas informações sobre o aplicativo.

**Figura 21 - Captura da Tela Configurações**



**Fonte: Elaborado pelos autores**

A tela de “Configurações”, apresentada na Figura 21, é composta por:

- **Botão voltar:** Para voltar para tela *home*.
- **Botão deslogar:** Para o usuário sair de sua conta.
- **Botão editar perfil:** Para o usuário ir para tela editar perfil.
- **Botão sincronizar calendário:** Para fazer uma sincronização dos eventos do calendário do aplicativo com o calendário do dispositivo.
- **Botão Fale conosco:** Para direcionar até o e-mail pessoal do usuário onde ele pode enviar uma mensagem para os desenvolvedores.

- **Botão política de privacidade:** Para mostrar a política de privacidade do aplicativo.
- **Botão sobre:** Para mostrar informações técnicas sobre o aplicativo

A Figura 22 apresenta a tela “Editar perfil”, no qual o usuário consegue alterar sua foto de perfil, seu nome, *e-mail* e senha.

**Figura 22 - Captura da tela Editar perfil**



**Fonte: Elaborado pelos autores**

A tela “Editar perfil”, apresentada na Figura 22, é composta por:

- **Botão voltar:** Para voltar para a tela de configurações.
- **Botão foto:** Para alterar a foto de perfil do usuário.
- **Campo editar nome:** Para alterar o nome do usuário.
- **Campo editar e-mail:** Para alterar o e-mail do usuário.
- **Botão alterar senha:** Para redirecionar até a tela alterar senha.

- **Botão deletar conta:** Para deletar sua conta.
- **Botão salvar alterações:** Para salvar todas as modificações que foram feitas.

A Figura 23 apresenta a tela “Alterar senha”, para o usuário modificar sua senha.

**Figura 23 - Captura da tela Alterar senha**

A imagem mostra a interface de usuário para alterar a senha. O cabeçalho da tela é azul com o texto "Alterar senha" e um ícone de seta para trás. Abaixo, há três campos de entrada brancos com bordas arredondadas, rotulados "Senha atual", "Nova senha" e "Confirmar nova senha". No canto inferior direito, há um botão azul com o ícone de um disco e o texto "Salvar Alterações".

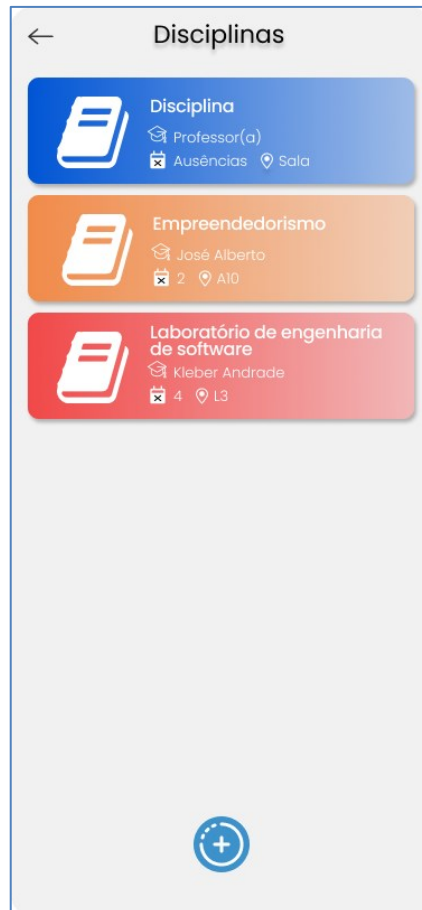
**Fonte: Elaborado pelos autores**

A tela “Alterar senha”, apresentada na Figura 23, é composta por:

- **Campo senha atual:** Para colocar a senha atual que está sendo usada.
- **Campo nova senha:** Para colocar uma nova senha.
- **Campo confirmar senha:** Para colocar novamente a nova senha.

A Figura 24 apresenta a tela “Disciplinas”, no qual o usuário consegue visualizar todas as disciplinas que ele adicionou, e também realizar adição de outras.

**Figura 24 - Captura da Tela de Disciplinas**



**Fonte: Elaborado pelos autores**

A tela “Disciplinas”, apresentada na Figura 24, é composta por:

- **Botão voltar:** Para direcionar à tela “Home”.
- **Card Disciplina:** Para direcionar a tela que contém mais informações sobre a disciplina.
- **Float Button:** Para adicionar uma disciplina.

A Figura 25 apresenta a tela “Adicionar Disciplina”, na qual o usuário pode fazer a adição de uma disciplina.

**Figura 25 - Captura da Tela adicionar disciplina**

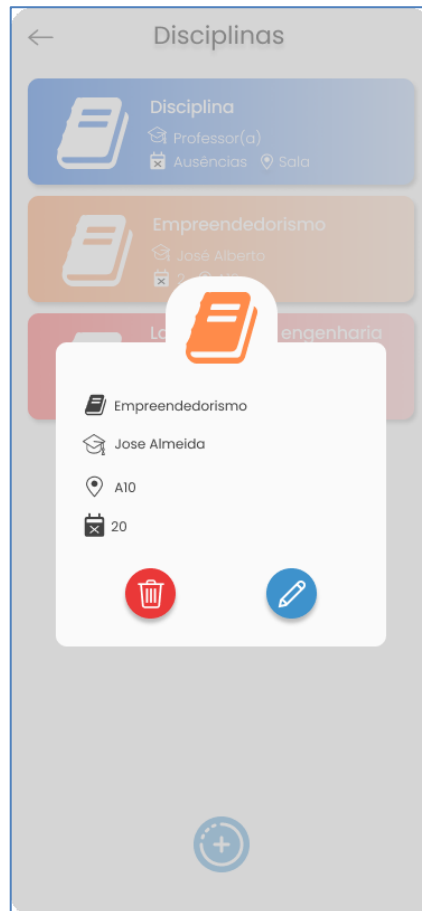


**Fonte: Elaborado pelos autores**

A tela “Adicionar Disciplina”, apresentada na Figura 25, é composta por:

- **Campo nome:** Para colocar o nome da disciplina.
- **Campo professor:** Para colocar o nome do professor da disciplina.
- **Campo sala:** Para colocar a sala.
- **Campo máximo de ausências:** Para colocar o máximo de ausências
- **Botão adicionar cor:** Para colocar uma cor na disciplina.
- **Botão Cancelar:** Para cancelar a adição da disciplina.
- **Botão Confirmar:** Para confirmar a adição da disciplina.

A Figura 26 apresenta a opção de remoção ou edição da disciplina na tela “Disciplina”, na qual o usuário consegue deletar ou editar a mesma.

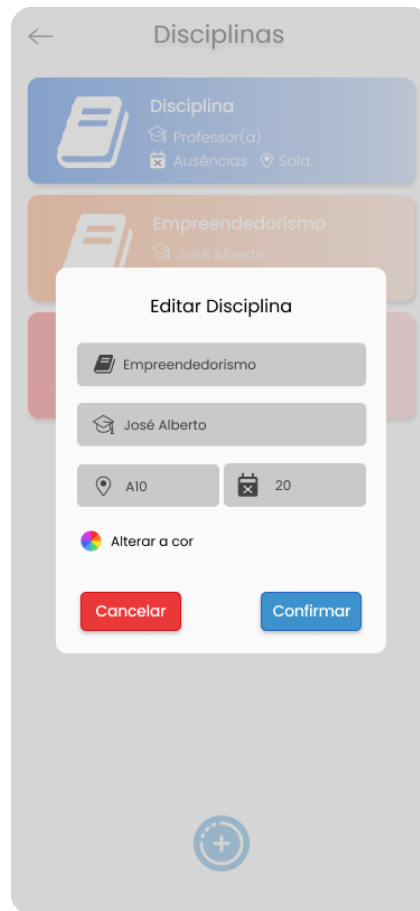
**Figura 26 - Captura da tela Disciplina**

**Fonte: Elaborado pelos autores**

A tela “Disciplina”, apresentada na Figura 26, é composta por:

- **Botão Deletar:** Para direcionar à tela deletar disciplina.
- **Botão Editar:** Para direcionar à tela editar disciplina.

A Figura 27 apresenta a tela “Editar disciplina”, na qual o usuário consegue alterar quaisquer informações daquela disciplina.

**Figura 27 - Captura da tela Editar Disciplina**

**Fonte: Elaborado pelos autores**

A tela “Editar Disciplina”, apresentada na Figura 27, é composta por:

- **Campo editar nome:** Para alterar o nome da disciplina.
- **Campo editar nome do professor:** Para alterar o nome do professor da disciplina.
- **Campo editar sala:** Para alterar a sala.
- **Campo editar máximo de ausências:** Para alterar a quantidade máxima de ausências permitida.
- **Botão editar cor:** Para alterar a cor da disciplina.
- **Botão Cancelar:** Para cancelar a edição da disciplina.
- **Botão confirmar:** Para confirmar a edição da disciplina.

A Figura 28 apresenta a opção de confirmação de remoção de disciplina, na tela “Disciplina”, na qual o usuário consegue confirmar ou não a remoção de disciplina escolhida anteriormente na tela “Disciplina”

**Figura 28 - Captura da tela Deletar Disciplina**



**Fonte: Elaborado pelos autores**

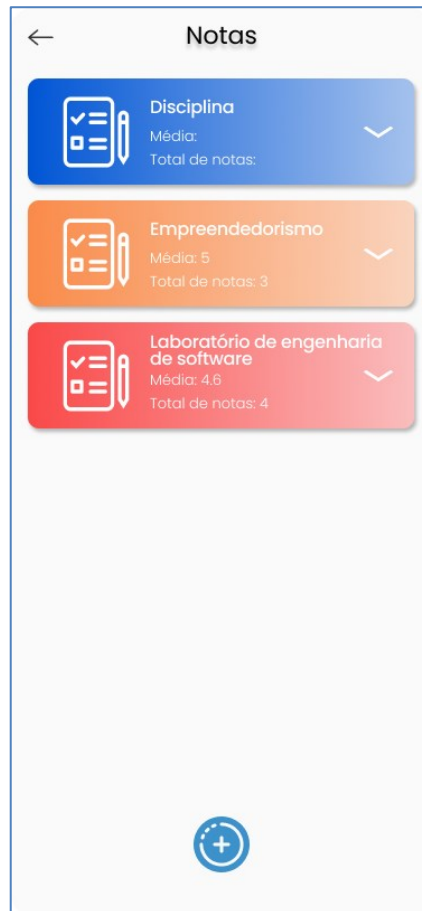
A tela “Deletar Disciplina”, apresentada na Figura 28, é composta por:

- **Botão Cancelar:** Para cancelar a eliminação da disciplina.
- **Botão Confirmar:** Para confirmar a eliminação da disciplina.

A Figura 29 apresenta a tela “Notas”, na qual o usuário consegue visualizar todas as disciplinas adicionadas anteriormente.



**Figura 29 - Captura da tela Notas.**



**Fonte: Elaborado pelos autores**

A tela “Notas”, apresentada na Figura 29, é composta por:

- **Botão voltar:** Para direcionar à tela “Home”.
- **Card disciplina:** Para direcionar ao *dropdown* das notas da disciplina.
- **Float Button** (ícone “⊕”): Para adicionar uma nota.

A Figura 30 apresenta a tela “Adicionar nota”, na qual o usuário consegue adicionar uma nota para uma disciplina já cadastrada.

**Figura 30 - Captura da tela Adicionar nota**

**Fonte: Elaborado pelos autores**

A tela “Adicionar nota”, apresentada na Figura 30, é composta por:

- **Campo título:** Para colocar o nome da nota.
- **Campo nota:** Para inserir o valor da nota.
- **Campo disciplina:** Para optar pela disciplina a qual a nota será adicionada (ao clicar será aberta uma tela de seleção com as disciplinas cadastradas).
- **Campo data:** Para colocar a data em que a atividade foi realizada (ao clicar será aberta uma tela de calendário).
- **Botões radio:** Para selecionar a categoria da nota, se é uma atividade ou prova.
- **Botão Cancelar:** Para cancelar a adição da nota.
- **Botão Confirmar:** Para confirmar a adição da nota.

A Figura 31 apresenta a opção *dropdown* das notas na tela “Notas”, na qual o usuário consegue visualizar todas as notas que adicionou para aquela disciplina, e também consegue clicar na nota para saber mais informações sobre ela.

**Figura 31 - Captura da tela Dropdown das notas**



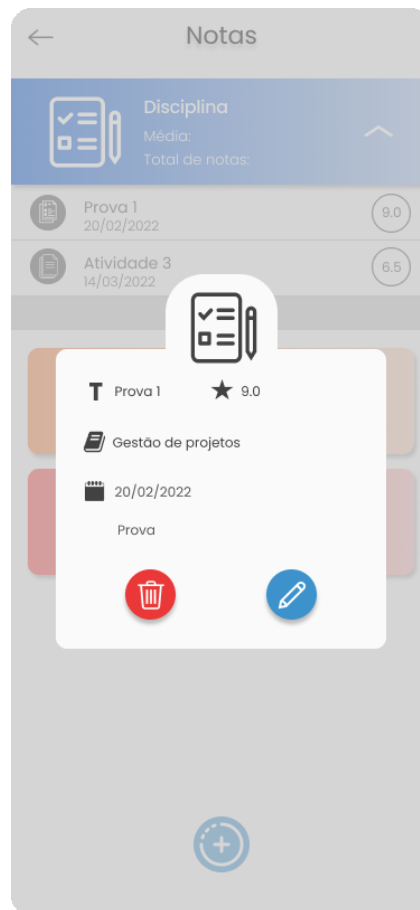
**Fonte: Elaborado pelos autores**

A tela “*Dropdown* das notas”, apresentada na Figura 31, é composta por:

- **Card disciplina:** Ao clicar são mostradas as notas que foram adicionadas aquela disciplina.

A Figura 24 apresenta a opção de remoção ou edição da nota na tela "Notas", na qual o usuário consegue visualizar todas as informações daquela nota, tal como seu título, nota, disciplina, data e tipo de tarefa, além de permitir sua remoção ou alteração sobre ela.

**Figura 32 - Captura da tela Notas**



**Fonte: Elaborado pelos autores**

A tela "Notas", apresentada na Figura 32, é composta por:

- **Botão Deletar:** Para direcionar à tela deletar a nota.
- **Botão Editar:** Para direcionar à tela editar a nota.

A Figura 33 apresenta a tela “Editar nota”, na qual o usuário pode editar quaisquer informações daquela nota.

**Figura 33 - Captura da tela Editar nota**



**Fonte: Elaborado pelos autores**

A tela “Editar nota”, apresentada na Figura 33, é composta por:

- **Campo título:** Para alterar o título da nota.
- **Campo valor da nota:** Para alterar o valor da nota.
- **Campo data:** Para editar a data em que a atividade foi realizada.
- **Botões radio:** Para selecionar outra categoria de nota.
- **Botão Cancelar:** Para cancelar a edição da nota.
- **Botão confirmar:** Para confirmar a edição da nota.

A Figura 34 apresenta a opção “deletar nota” na tela “Nota”, na qual o usuário consegue eliminar determinada nota.

**Figura 34 - Captura da tela com a opção para deletar nota**

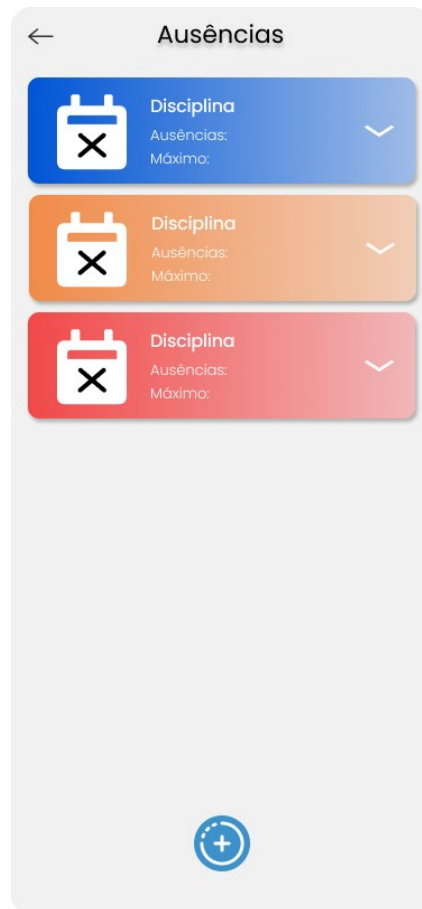


**Fonte: Elaborado pelos autores**

A tela de opção “Deletar nota”, da tela “Nota”, apresentada na Figura 34, é composta por:

- **Botão Cancelar:** Para cancelar a eliminação da nota.
- **Botão Confirmar:** Para confirmar a eliminação da nota.

A Figura 35 apresenta a tela “Ausências”, na qual o usuário consegue visualizar todas as disciplinas cadastradas, de forma a permitir a inserção da quantidade de faltas cometidas na disciplina escolhida entre as cadastradas.

**Figura 35 - Captura da tela Ausências**

Fonte: Elaborado pelos autores

A tela “Ausências”, apresentada na Figura 35, é composta por:

- **Botão voltar:** Para direcionar à tela “Home”.
- **Card Disciplina:** Para direcionar à tela *dropdown* de ausências que contém as ausências cometidas na disciplina, bem como permite a inserção e atualização de quantas faltas o aluno cometeu.
- **Float Button:** Para adicionar uma ausência.

A Figura 36 apresenta a tela “Adicionar ausência”, na qual o usuário consegue adicionar uma ausência para uma disciplina cadastrada.

**Figura 36 - Captura da tela “Adicionar ausência”**



**Fonte: Elaborado pelos autores**

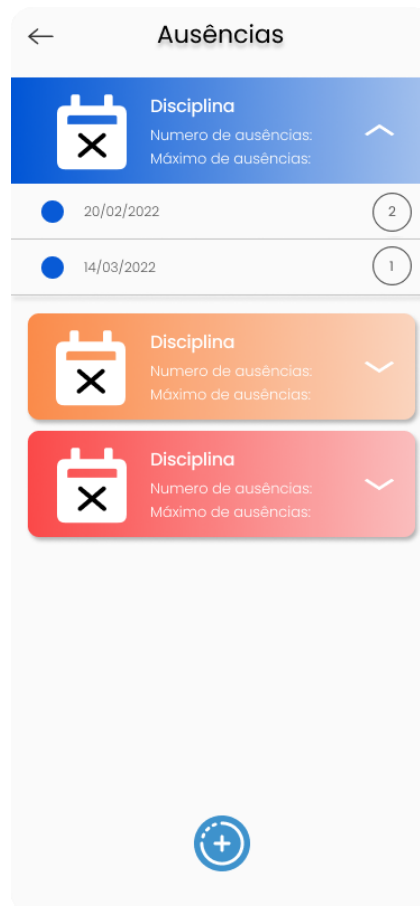
A tela “Adicionar ausência”, apresentada na Figura 36, é composta por:

- **Botão *range slider*:** Para informar a quantidade de ausências.
- **Campo disciplina:** Para colocar a disciplina daquela ausência.
- **Campo data:** Para colocar a data da ausência (ao clicar será aberta uma tela de calendário).
- **Botão Cancelar:** Para cancelar a adição da ausência.
- **Botão Confirmar:** Para confirmar a adição da ausência.



A Figura 37 apresenta uma tela com *dropdown* de ausências da tela “Ausências”, na qual o usuário consegue visualizar todas as ausências daquela disciplina.

**Figura 37 - Captura da tela *Dropdown* de ausências**



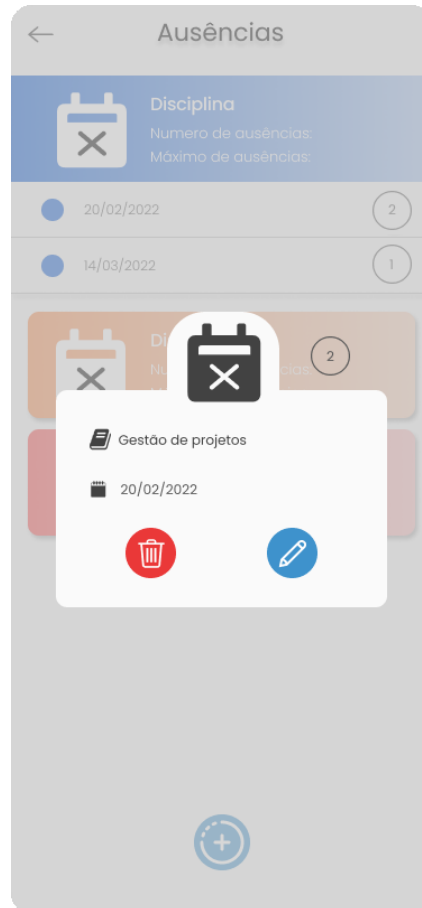
**Fonte: Elaborado pelos autores**

A tela “*dropdown* de ausências”, apresentada na Figura 35, é composta por:

- **Card disciplina:** Ao clicar são mostradas as ausências que foram adicionadas para aquela disciplina.

A Figura 38 apresenta a tela "Ausências", na qual o usuário consegue visualizar informações sobre aquela ausência, e também com a opção de deletar ou editar a ausência.

**Figura 38 - Captura da ausência**



**Fonte: Elaborado pelos autores**

A tela "Ausências", com informações sobre a ausência em questão, apresentada na Figura 38, é composta por:

- **Botão deletar:** Para deletar a ausência.
- **Botão editar:** Para editar a ausência.

A Figura 39 apresenta a tela “Editar ausência”, na qual o usuário pode editar as informações daquela ausência.

**Figura 39 - Captura da tela Editar ausência**



**Fonte: Elaborado pelos autores**

A tela “Editar ausência”, apresentada na Figura 37 é composta por:

- **Botão *range slider*:** Para editar a quantidade de ausências.
- **Campo disciplina:** Para editar a disciplina daquela ausência.
- **Campo data:** Para editar a data da ausência (ao clicar será aberta uma tela de calendário).
- **Botão Cancelar:** Para cancelar a edição da ausência.
- **Botão Confirmar:** Para confirmar a edição da ausência.

A Figura 40 apresenta a tela com a opção para deletar ausência, da tela “Ausências”, na qual o usuário consegue fazer a eliminação da ausência.

**Figura 40 - Captura da tela de opção para deletar ausência, na tela “Ausências”.**



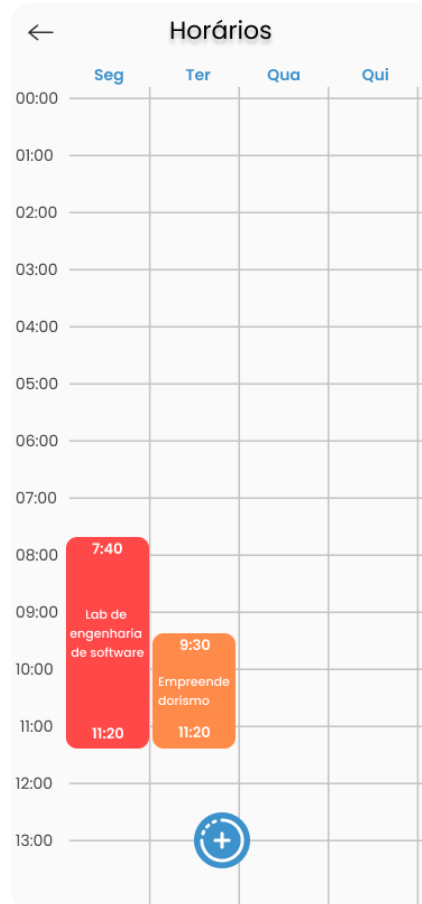
**Fonte: Elaborado pelos autores**

A tela com a opção para deletar ausência, da tela “Ausências”, apresentada na Figura 40, é composta por:

- **Botão Cancelar:** Para cancelar a eliminação da ausência.
- **Botão confirmar:** Para confirmar a eliminação da ausência.

A Figura 41 apresenta a tela "Horários", na qual o usuário consegue visualizar o horário das aulas daquele dia.

**Figura 41 - Captura da tela Horários.**



**Fonte: Elaborado pelos autores**

A tela "Horários", apresentada na Figura 41, é composta por:

- **Grid dos horários:** Para mostrar os horários das aulas daquele dia.
- **Card:** Para mostrar um *card* do horário cadastrado.
- **Float button:** Para adicionar um horário.

A Figura 42 apresenta a tela de opção para adicionar horário, na tela “Horários”, na qual o usuário pode adicionar um horário para uma disciplina que já foi criada anteriormente.

**Figura 42 - Captura da tela adicionar horário**



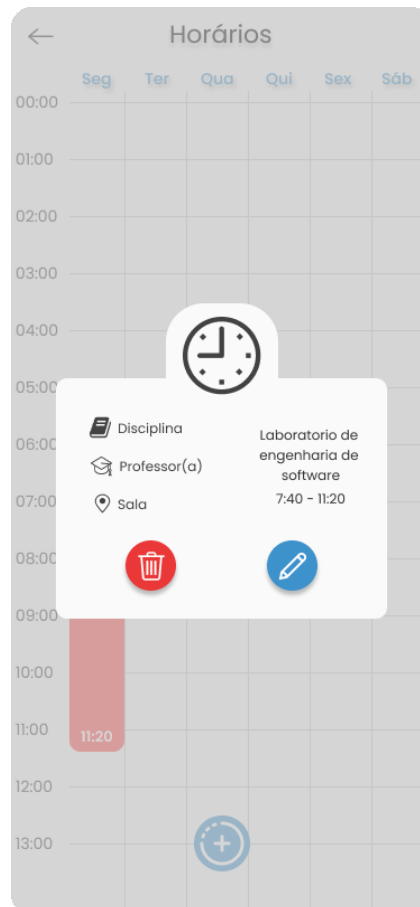
**Fonte: Elaborado pelos autores**

A tela com a opção para adicionar horário, na tela “Horários”, apresentada na Figura 42, é composta por:

- **Campo disciplina:** Para selecionar a disciplina daquele horário.
- **Select dia:** Para selecionar o dia da semana em que aquela disciplina ocorre.
- **Campo início:** Para colocar o horário de início daquela aula.
- **Campo fim:** Para colocar o horário de término daquela aula.
- **Select notificação:** Para selecionar se o usuário quer ser notificado quando a aula começar.
- **Botão cancelar:** Para cancelar a adição do horário.
- **Botão confirmar:** Para confirmar a adição do horário.

A Figura 43 apresenta a tela com informações sobre um horário escolhido, com a finalidade de remoção ou alteração, na tela “Horários”, na qual o usuário consegue visualizar mais informações de um horário cadastrado.

**Figura 43 - Captura da tela Horários.**



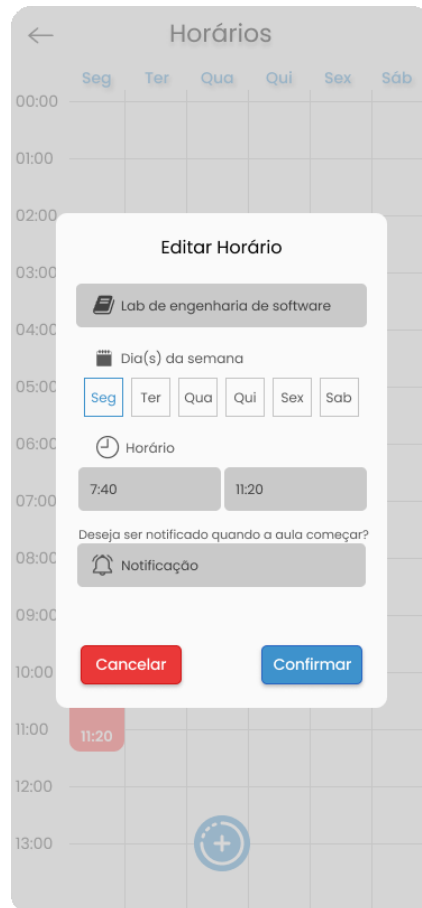
**Fonte: Elaborado pelos autores**

A tela com as opções para editar nota, na tela “Horários”, apresentada na Figura 43, é composta por:

- **Botão deletar:** Para deletar o horário.
- **Botão editar:** Para editar o horário.

A Figura 42 apresenta a tela de opção para editar horário, na tela “Horários”, na qual o usuário consegue editar as informações de um horário escolhido entre os cadastrados.

**Figura 44 - Captura da tela com a opção para editar horário, da tela Horários.**



**Fonte: Elaborado pelos autores**

A tela com a opção para editar horário, da tela “Horários”, apresentada na Figura 44, é composta por:

- **Campo disciplina:** Para selecionar a disciplina daquele horário.
- **Select dia:** Para selecionar o dia da semana em que aquela disciplina ocorre.
- **Campo início:** Para colocar o horário de início daquela aula.
- **Campo fim:** Para colocar o horário de término daquela aula.
- **Select notificação:** Para selecionar se o usuário quer ser notificado quando a aula começar.
- **Botão cancelar:** Para cancelar a edição do horário.



- **Botão confirmar:** Para confirmar a adição do horário.

A Figura 45 apresenta a tela “Horário”, com a opção para deletar horário, na qual o usuário pode eliminar um horário criado anteriormente.

**Figura 45 - Captura da Deletar horário**



**Fonte: Elaborado pelos autores**

A tela com a opção deletar horário, da tela “Horário”, apresentada na Figura 45, é composta por:

- **Botão cancelar:** Para cancelar a eliminação do horário.
- **Botão confirmar:** Para confirmar a eliminação do horário.

A Figura 46 apresenta a tela "Agenda", na qual o usuário consegue visualizar e adicionar eventos, entre a data de uma prova ou de uma atividade.

**Figura 46 - Captura da tela Agenda**



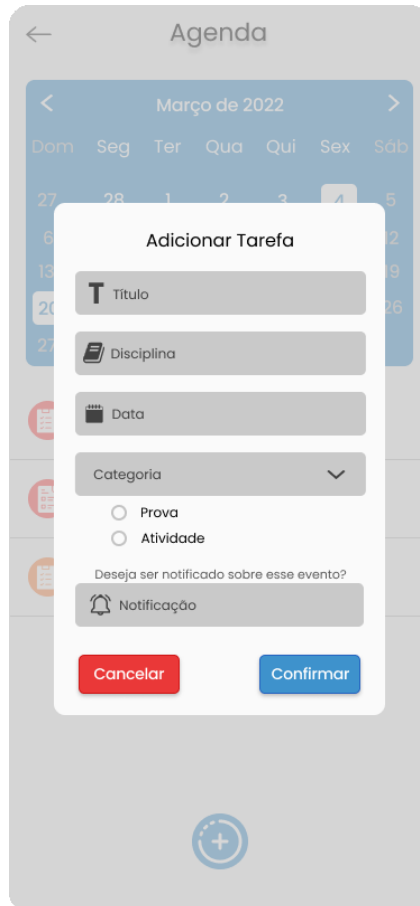
**Fonte: Elaborado pelos autores**

A tela "Agenda", apresentada na Figura 46, é composta por:

- **Botão voltar (←):** Para voltar para a tela *Home*.
- **Calendário:** Para mostrar o dia dos eventos já adicionados.
- **Card evento:** Para mostrar informações de um evento.
- **Float button:** Para adicionar um evento.

A Figura 47 apresenta a tela com a opção para adicionar evento, da tela “Agenda”, na qual o usuário consegue inserir informações de um evento para ser adicionado.

**Figura 47 - Captura da tela com a opção para adicionar evento na tela “Agenda”**



**Fonte: Elaborado pelos autores**

A tela com a opção para adicionar evento, da tela “Agenda”, apresentada na Figura 47, é composta por:

- **Campo título:** Para adicionar o título do evento.
- **Campo disciplina:** Para escolher a disciplina daquele evento.
- **Campo data:** Para optar pela data em que aquele evento vai terminar.
- **Botões *radio*:** Para selecionar a categoria do evento.
- **Botão cancelar:** Para cancelar a adição do evento.
- **Botão confirmar:** Para confirmar a adição do evento.

A Figura 48 apresenta a tela com a opção para remoção ou edição de evento, na tela “Agenda”, na qual o usuário consegue visualizar mais informações daquele evento.

**Figura 48 - Captura da tela com a opção evento na tela Agenda**



**Fonte: Elaborado pelos autores**

A tela com a opção para remoção ou edição de evento, da tela “Agenda”, apresentada na Figura 48, é composta por:

- **Botão deletar:** Para deletar o evento.
- **Botão editar:** Para editar o evento.

A Figura 49 apresenta a tela com a opção para editar evento, da tela “Agenda”, na qual o usuário consegue editar as informações de um evento adicionado anteriormente.

**Figura 49 - Captura da tela editar evento.**



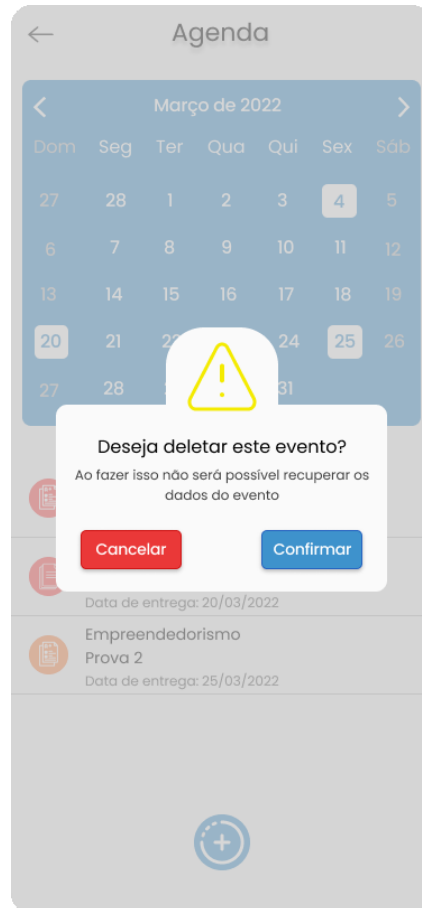
**Fonte: Elaborado pelos autores**

A tela “Editar evento”, apresentada na Figura 49, é composta por:

- **Campo título:** Para alterar o título do evento.
- **Campo disciplina:** Para alterar a disciplina do evento.
- **Campo data:** Para editar a data do evento.
- **Botões radio:** Para alterar a categoria do evento.
- **Botão cancelar:** Para cancelar a edição do evento.
- **Botão confirmar:** Para confirmar a edição do evento.

A Figura 48 apresenta a tela com a opção para deletar evento, da tela “Agenda”, na qual o usuário consegue deletar um evento adicionado anteriormente.

**Figura 50 - Captura da deletar evento**



**Fonte: Elaborado pelos autores**

A tela com a opção para deletar evento, da tela “Agenda”, apresentada na Figura 50, é composta por:

- **Botão cancelar:** Para cancelar a eliminação do evento.
- **Botão confirmar:** Para confirmar a eliminação do evento.

## 4. MODELAGEM

Na fase da modelagem é feita a documentação do aplicativo, se trata de diagramas que facilitam na compreensão do projeto de forma padronizada. A documentação deste trabalho utilizará a linguagem de modelagem *Unified Modeling Language* ou Linguagem Unificada de Modelagem (UML) que é uma linguagem padrão para modelagem e documentação de sistemas orientados a objetos.

### 4.1. UML: Diagramas de Casos de Uso

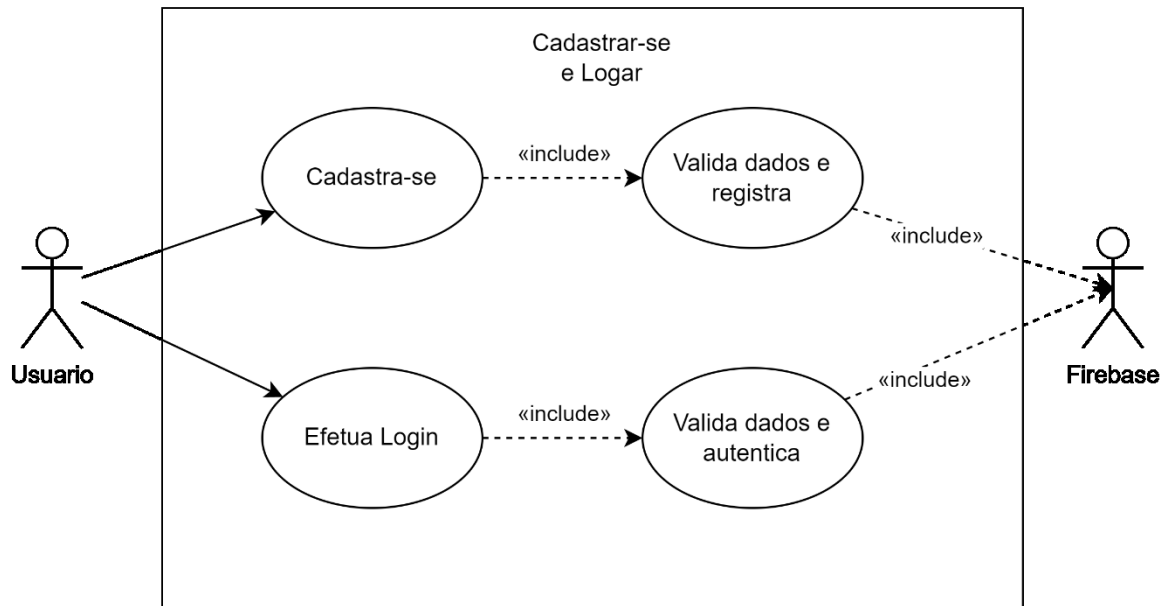
Diagrama de caso de uso basicamente são um resumo dos detalhes dos usuários/autores do sistema e também as interações deles com o sistema, ele é composto por um conjunto de símbolos e conectores especializados, tais como caixas, pacotes, associações, bonecos palito e entre outros. Em suma se utiliza um diagrama de caso de uso para ter uma visão geral do relacionamento entre casos de uso, atores e sistemas.

Os atores que interagem com o sistema são: o Usuário e *Firebase*. O sistema é um caso de uso explícito e se trata do sistema em si em que os casos de uso acontecem.

- **Usuário** é o ator que representa os utilizadores deste aplicativo. Um ator pode, por exemplo, manter disciplinas, manter notas, manter ausências, manter eventos.
- **Firebase** representa o banco de dados em tempo real, onde o sistema armazena as informações de cadastro de cada usuário.
- **SQFLite** representa o banco de dados local, onde o sistema armazena as informações de cada disciplina e de todos os dados vinculados a ela.

A Figura 50 apresenta o caso de uso para a entrada e cadastro do usuário no sistema.

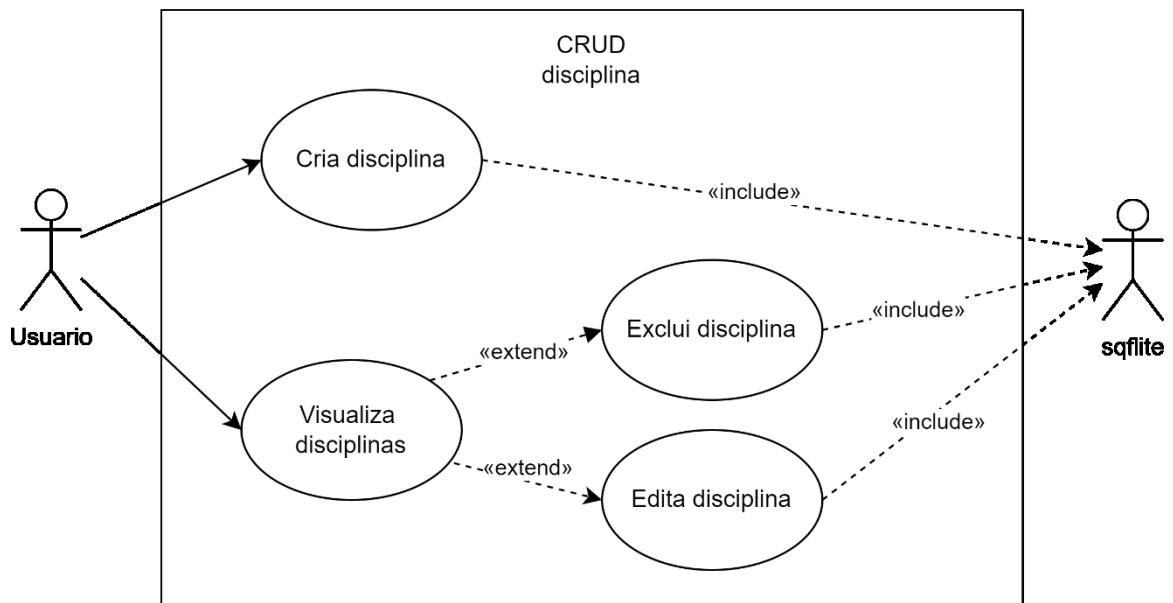
**Figura 51 – Diagrama de caso de uso do cadastro e login do usuário**



Fonte: Elaborado pelos autores .

A Figura 52 apresenta o caso de uso para o CRUD de disciplinas do sistema.

**Figura 52 – Diagrama de caso de uso do CRUD de disciplinas**

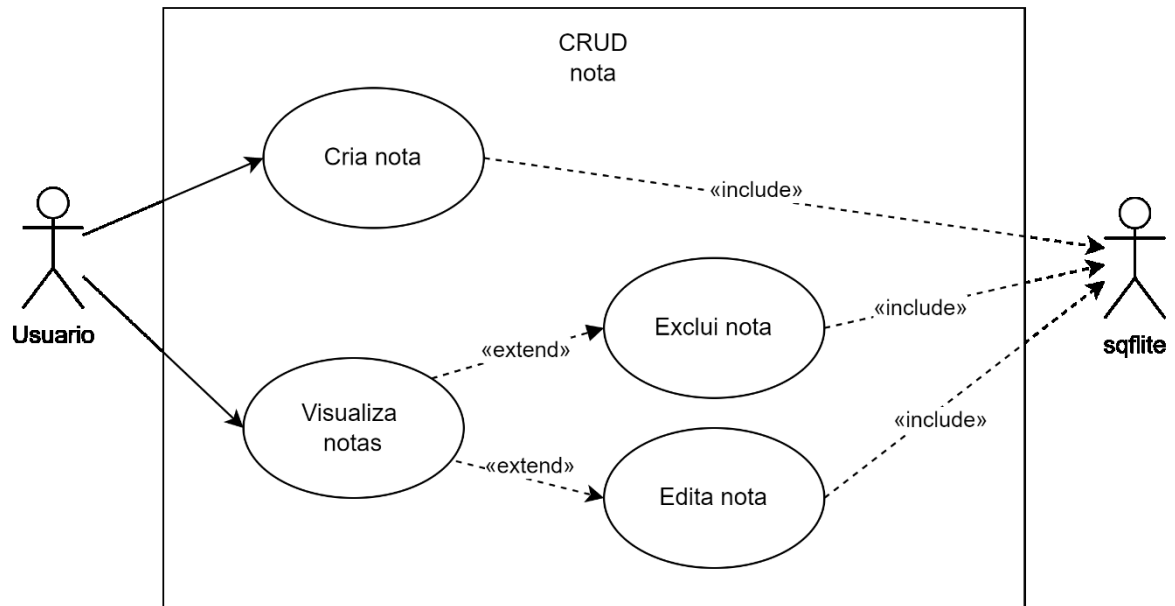


Fonte: Elaborado pelos autores.

A Figura 53 apresenta o caso de uso para o CRUD de notas do sistema.



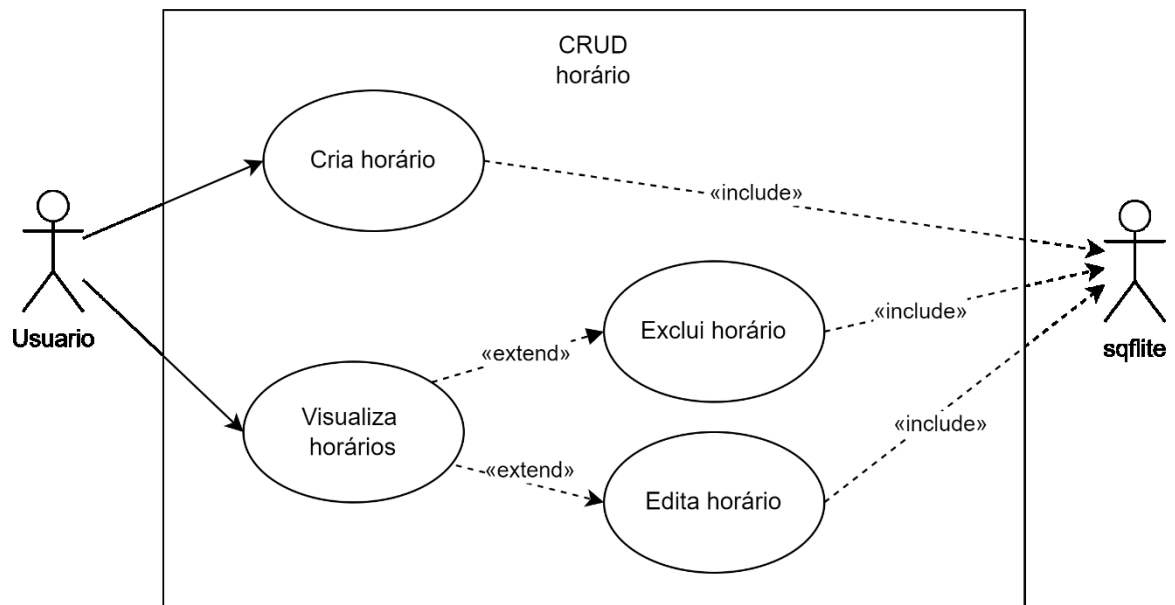
**Figura 53 – Diagrama de Caso de Uso do CRUD de Notas**



Fonte: Elaborado pelos autores.

A Figura 54 apresenta o caso de uso para o CRUD de horários do sistema.

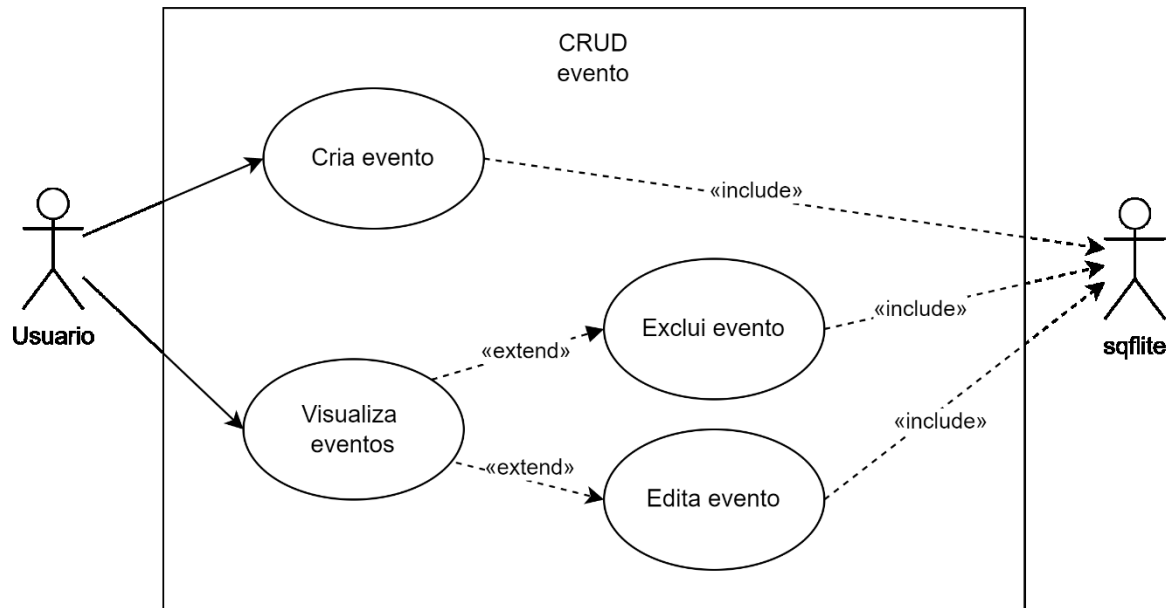
**Figura 54 - Diagrama de Caso de Uso do CRUD de Horários**



Fonte: Elaborado pelos autores.

A Figura 55 apresenta o caso de uso para o CRUD de eventos do sistema.

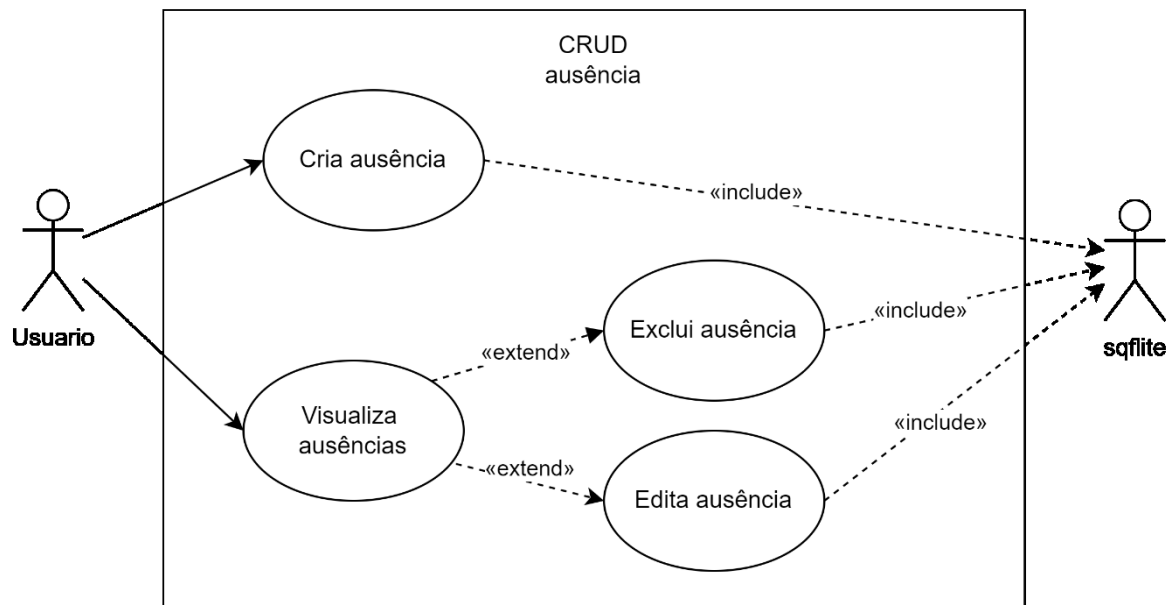
**Figura 55 – Diagrama de Caso de Uso do CRUD de Eventos**



Fonte: Elaborado pelos autores.

A Figura 56 apresenta o caso de uso para o CRUD de ausências do sistema.

**Figura 56 – Diagrama de Caso de Uso do CRUD de Ausências**



Fonte: Elaborado pelos autores.

#### 4.1.1. Documentação dos Casos de Uso

Cada funcionalidade dos diagramas de casos de uso será descrita da Tabela 18 à Tabela 23.

A Tabela 18 descreve a documentação para o caso de uso “Entrar no Sistema / Cadastrar Usuário”.

**Tabela 18 – Caso de uso “Entrar no Sistema / Cadastrar Usuário”.**

<b>Nome do caso de uso</b>	Entrar no Sistema / Cadastrar Usuário
<b>Atores envolvidos</b>	Usuário, <i>Firebase</i> .
<b>Objetivo</b>	Este caso de uso descreve os passos do <i>login</i> e cadastro de um usuário no sistema
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário pode clicar em cadastrar-se inserir todos os dados pedidos pelo sistema	
	2. O <i>Firebase</i> cria um usuário com um id único com as informações inseridas
3. O usuário clica em fazer <i>login</i> insere o seu <i>e-mail</i> e senha	
	4. O <i>firebase</i> efetua o <i>login</i> com as credenciais do usuário.
<b>Validações</b>	No <i>login</i> : verifica se o usuário existe, e se as credenciais estão corretas No cadastro: Existe um <i>regex</i> para verificar se o <i>e-mail</i> é válido, um para saber se a senha possui no mínimo oito caracteres, dentre eles pelo menos três numéricos, três letras e dois símbolos, também possui um <i>regex</i> para uma validação utilizando dois campos para verificar se as senhas são iguais

**Fonte: Elaborado pelo autor.**

A Tabela 19 descreve a documentação para o caso de uso “Inserir / Editar / Deletar Disciplina”.

**Tabela 19 – Caso de uso “Inserir / Editar / Deletar Disciplina”.**

<b>Nome do caso de uso</b>	Inserir / Editar Disciplina
<b>Atores envolvidos</b>	Usuário, <i>SQFLite</i> .

<b>Objetivo</b>	Este caso de uso descreve os passos de inserção e edição e exclusão de disciplinas
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em disciplinas na tela inicial	
	2. O sistema abre o módulo de disciplinas
3. O usuário pode clicar no botão de adicionar disciplina	
	5. O sistema abre um <i>dialog</i> onde o usuário pode inserir as informações da disciplina
6. O usuário insere as informações referente a disciplina e clica no botão “confirmar” para criar a disciplina	
	7. O sistema cria um novo registro no banco de dados na tabela disciplina com os dados fornecidos pelo usuário
8. O usuário clica em uma disciplina já existente listada na tela	
	9. O sistema abre um <i>dialog</i> onde o usuário pode visualizar as informações relacionadas a disciplina selecionada e escolher entre editar ou deletar a disciplina
10. O usuário clica em deletar disciplina	
	11. O sistema deleta a disciplina selecionada da tabela disciplina e todas as outras informações relacionadas a disciplina deletada em suas respectivas tabelas
12. O usuário clica em editar disciplina	
	13. O sistema abre um <i>dialog</i> onde o usuário pode ver e modificar as informações da disciplina selecionada e salvar as alterações
14. O usuário clica em salvar alterações	
	15. O sistema atualiza os dados inseridos no banco de dados referente a disciplina editada
<b>Validações</b>	Todos os campos devem ser preenchidos e cada um deles tem um limite de caracteres

**Fonte: Elaborado pelo autor.**

A Tabela 20 descreve a documentação para o caso de uso “Inserir / Modificar / Deletar Nota”.

**Tabela 20 – Caso de uso “Inserir / Modificar / Deletar Nota”.**

<b>Nome do caso de uso</b>	Inserir / Editar notas
<b>Atores envolvidos</b>	Usuário, <i>SQFlite</i> .
<b>Objetivo</b>	Este caso de uso descreve os passos de inserção e edição e exclusão de notas
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em notas na tela inicial	
	2. O sistema abre o módulo de notas
3. O usuário pode clicar no botão de adicionar nota	
	16. O sistema abre um <i>dialog</i> onde o usuário pode inserir as informações da nota
17. O usuário insere as informações referente a nota e clica no botão “confirmar” para criar a disciplina	
	18. O sistema cria um novo registro no banco de dados na tabela notas com os dados fornecidos pelo usuário
19. O usuário clica em uma nota já existente listada na tela	
	20. O sistema abre um <i>dialog</i> onde o usuário pode visualizar as informações relacionadas a nota selecionada e escolher entre editar ou deletar a nota
21. O usuário clica em deletar nota	
	22. O sistema deleta a nota selecionada da tabela notas
23. O usuário clica em editar nota	
	24. O sistema abre um <i>dialog</i> onde o usuário pode ver e modificar as informações da nota selecionada e salvar as alterações
25. O usuário clica em salvar alterações	
	26. O sistema atualiza os dados inseridos no banco de dados referente a nota editada

<b>Validações</b>	Todos os campos devem ser preenchidos e cada um deles tem um limite de caracteres
-------------------	---

**Fonte: Elaborado pelo autor.**

A Tabela 21 descreve a documentação para o caso de uso “Inserir /Modificar Horários”.

**Tabela 21 – Caso de uso “Inserir / Modificar Horários”.**

<b>Nome do caso de uso</b>	Inserir / Editar / Deletar Horários
<b>Atores envolvidos</b>	Usuário, <i>SQFlite</i> .
<b>Objetivo</b>	Este caso de uso descreve os passos de inserção e edição e exclusão de notas
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em horários na tela inicial	
	2. O sistema abre o módulo de horários
3. O usuário pode clicar no botão de adicionar horário	
	27. O sistema abre um <i>dialog</i> onde o usuário pode inserir as informações do horário
28. O usuário insere as informações referente ao horário e clica no botão “confirmar” para criar o horário	
	29. O sistema cria um novo registro no banco de dados na tabela horários com os dados fornecidos pelo usuário
30. O usuário clica em um horário já existente listado na tela	
	31. O sistema abre um <i>dialog</i> onde o usuário pode visualizar as informações relacionadas ao horário selecionado e escolher entre editar ou deletar o horário
32. O usuário clica em deletar horário	
	33. O sistema deleta o horário selecionado da tabela horários
34. O usuário clica em editar horário	

	35. O sistema abre um <i>dialog</i> onde o usuário pode ver e modificar as informações do horários selecionado e salvar as alterações
36. O usuário clica em salvar alterações	
	37. O sistema atualiza os dados inseridos no banco de dados referente ao horário editada
<b>Validações</b>	Todos os campos devem ser preenchidos e cada um deles tem um limite de caracteres

**Fonte: Elaborado pelo autor.**

A Tabela 22 descreve a documentação para o caso de uso “Inserir /Modificar Eventos”.

**Tabela 22 – Caso de uso “Inserir / Modificar Eventos”.**

<b>Nome do caso de uso</b>	Inserir / Editar / Deletar Eventos
<b>Atores envolvidos</b>	Usuário, <i>SQFlite</i> .
<b>Objetivo</b>	Este caso de uso descreve os passos de inserção e edição e exclusão de eventos
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em agenda na tela inicial	
	2. O sistema abre o módulo de agenda
3. O usuário pode clicar no botão de adicionar evento	
	38. O sistema abre um <i>dialog</i> onde o usuário pode inserir as informações do evento
39. O usuário insere as informações referente ao evento e clica no botão “confirmar” para criar o evento	
	40. O sistema cria um novo registro no banco de dados na tabela eventos com os dados fornecidos pelo usuário
41. O usuário clica em um evento já existente listado na tela	
	42. O sistema abre um <i>dialog</i> onde o usuário pode visualizar as informações relacionadas ao evento selecionado e escolher entre editar ou deletar o evento

43. O usuário clica em deletar evento	
	44. O sistema deleta o evento selecionado da tabela eventos
45. O usuário clica em editar evento	
	46. O sistema abre um <i>dialog</i> onde o usuário pode ver e modificar as informações do evento selecionado e salvar as alterações
47. O usuário clica em salvar alterações	
	48. O sistema atualiza os dados inseridos no banco de dados referente ao horário editada
<b>Validações</b>	Todos os campos devem ser preenchidos e cada um deles tem um limite de caracteres

Fonte: Elaborado pelo autor .

A Tabela 23 descreve a documentação para o caso de uso “Inserir /Modificar / Deletar Ausências”.

**Tabela 23 – Caso de uso “Inserir / Modificar / Deletar Ausências”.**

<b>Nome do caso de uso</b>	Inserir / Editar / Deletar Ausências
<b>Atores envolvidos</b>	Usuário, <i>SQFlite</i> .
<b>Objetivo</b>	Este caso de uso descreve os passos de inserção e edição e exclusão de ausências
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em ausências na tela inicial	
	2. O sistema abre o módulo de ausência
3. O usuário pode clicar no botão de adicionar ausência	
	49. O sistema abre um <i>dialog</i> onde o usuário pode inserir as informações da ausência
50. O usuário insere as informações referente a ausência e clica no botão “confirmar” para criar a ausência	



	51. O sistema cria um novo registro no banco de dados na tabela ausências com os dados fornecidos pelo usuário
52. O usuário clica em uma ausência já existente listado na tela	
	53. O sistema abre um <i>dialog</i> onde o usuário pode visualizar as informações relacionadas a ausência selecionada e escolher entre editar ou deletar a ausência
54. O usuário clica em deletar ausência	
	55. O sistema deleta a ausência selecionada da tabela ausências
56. O usuário clica em editar ausência	
	57. O sistema abre um <i>dialog</i> onde o usuário pode ver e modificar as informações da ausência selecionada e salvar as alterações
58. O usuário clica em salvar alterações	
	59. O sistema atualiza os dados inseridos no banco de dados referente ao horário editada
<b>Validações</b>	Todos os campos devem ser preenchidos e cada um deles tem um limite de caracteres

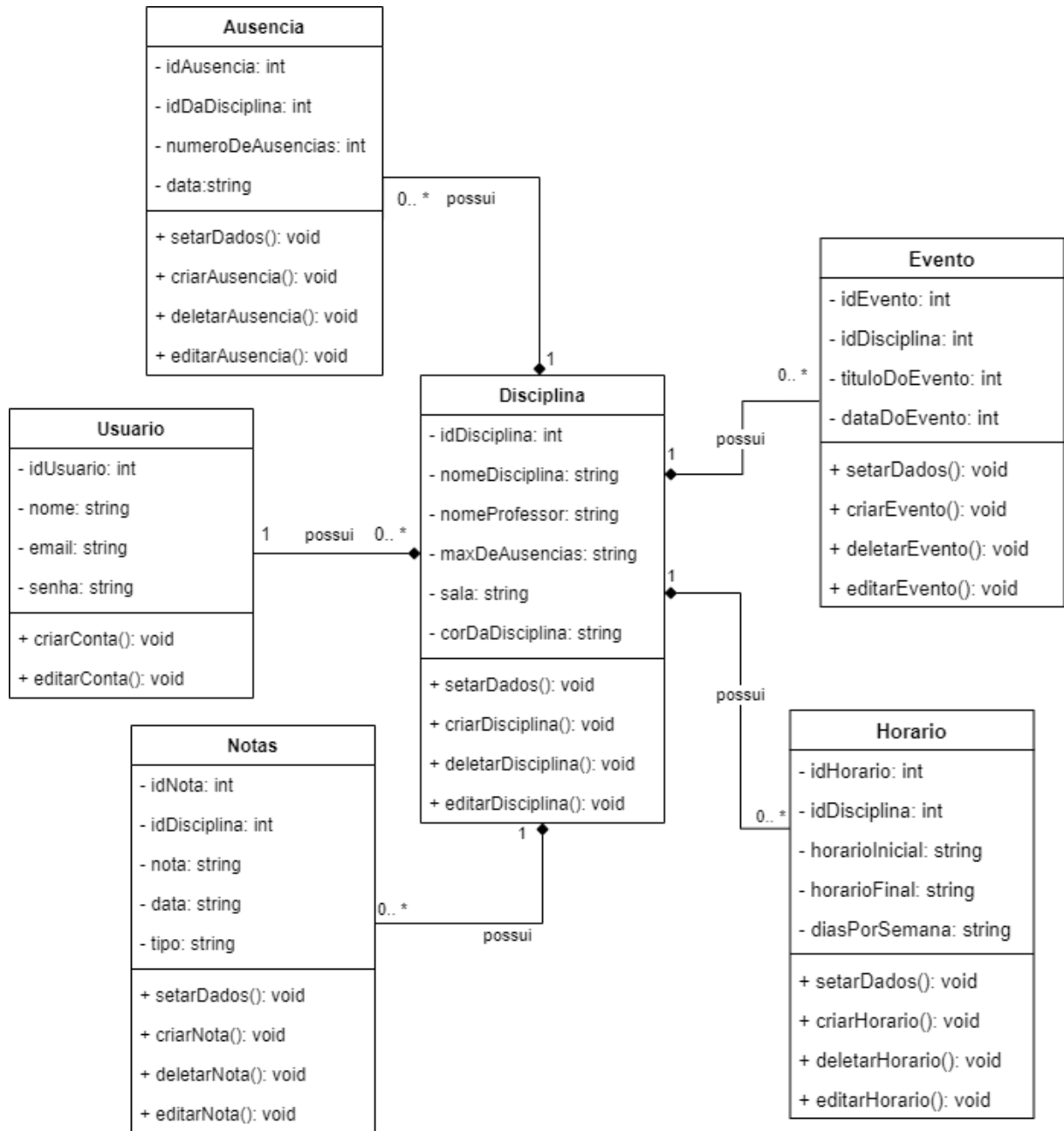
Fonte: Elaborado pelo autor .

## 4.2 UML: Diagrama de Classes

Diagramas de classes estão entre os tipos mais úteis de diagramas UML pois mapeiam de forma clara a estrutura de um determinado sistema ao modelar suas classes, seus atributos, operações e relações entre objetos.

A Figura 57 apresenta o diagrama de classes completo do aplicativo.

**Figura 57 – Diagrama de Classes do aplicativo**



Fonte: Elaborado pelos autores.

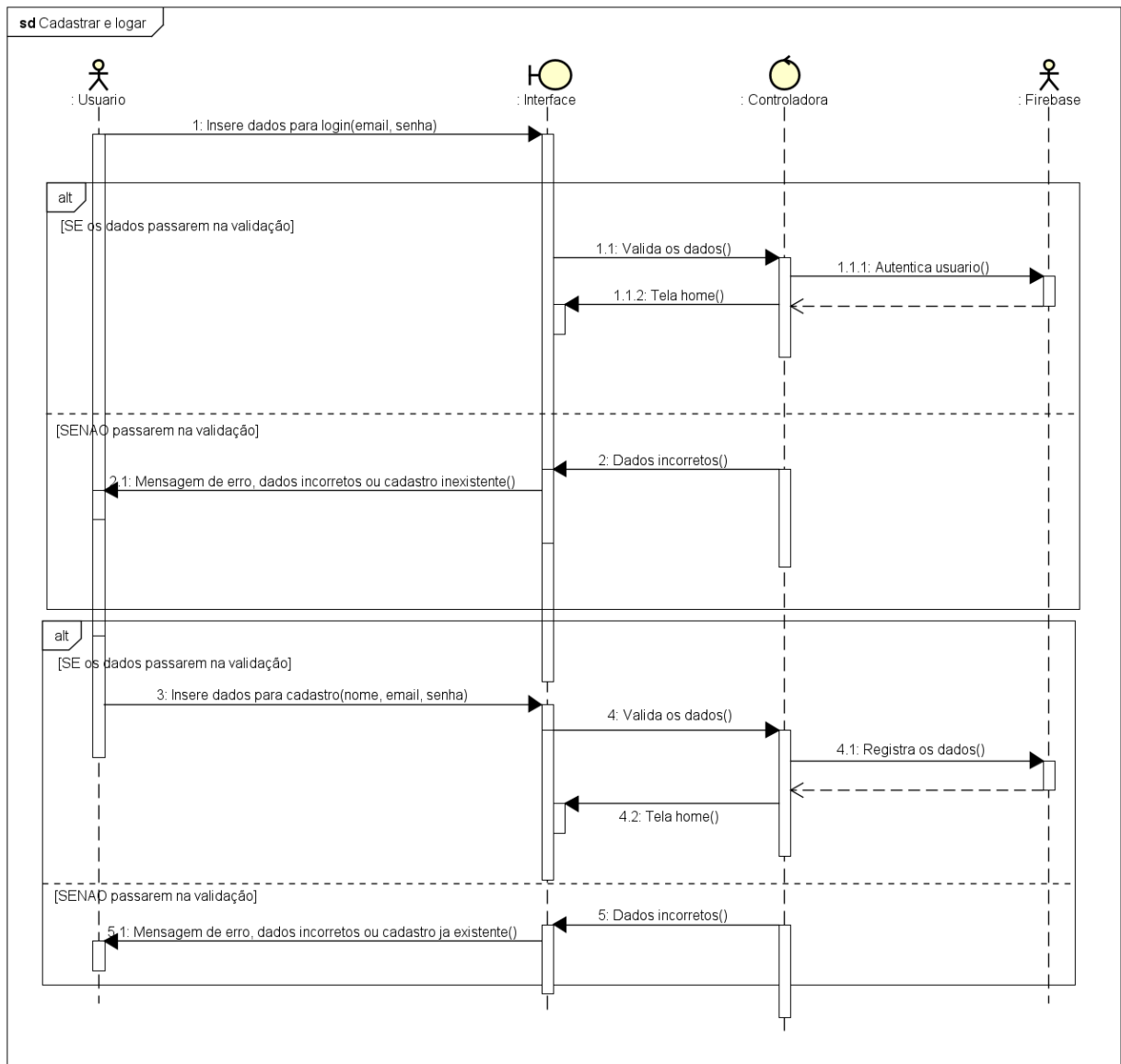
### 4.3. UML: Diagramas de Sequência

Diagrama de sequência é um diagrama usado em UML, evidenciando a sequência de processos em um sistema, é uma espécie de diagrama de interação, pois descreve como, e em qual ordem, um grupo de objetos trabalha em conjunto. Como um projeto pode ter uma grande quantidade de métodos em classes distintas, pode ser trabalhoso de se estabelecer a sequência global do comportamento.

Cada diagrama de sequência do aplicativo será apresentado da Figura 58 à Figura 63.

A Figura 58 apresenta o diagrama de sequência do cadastro e login do usuário.

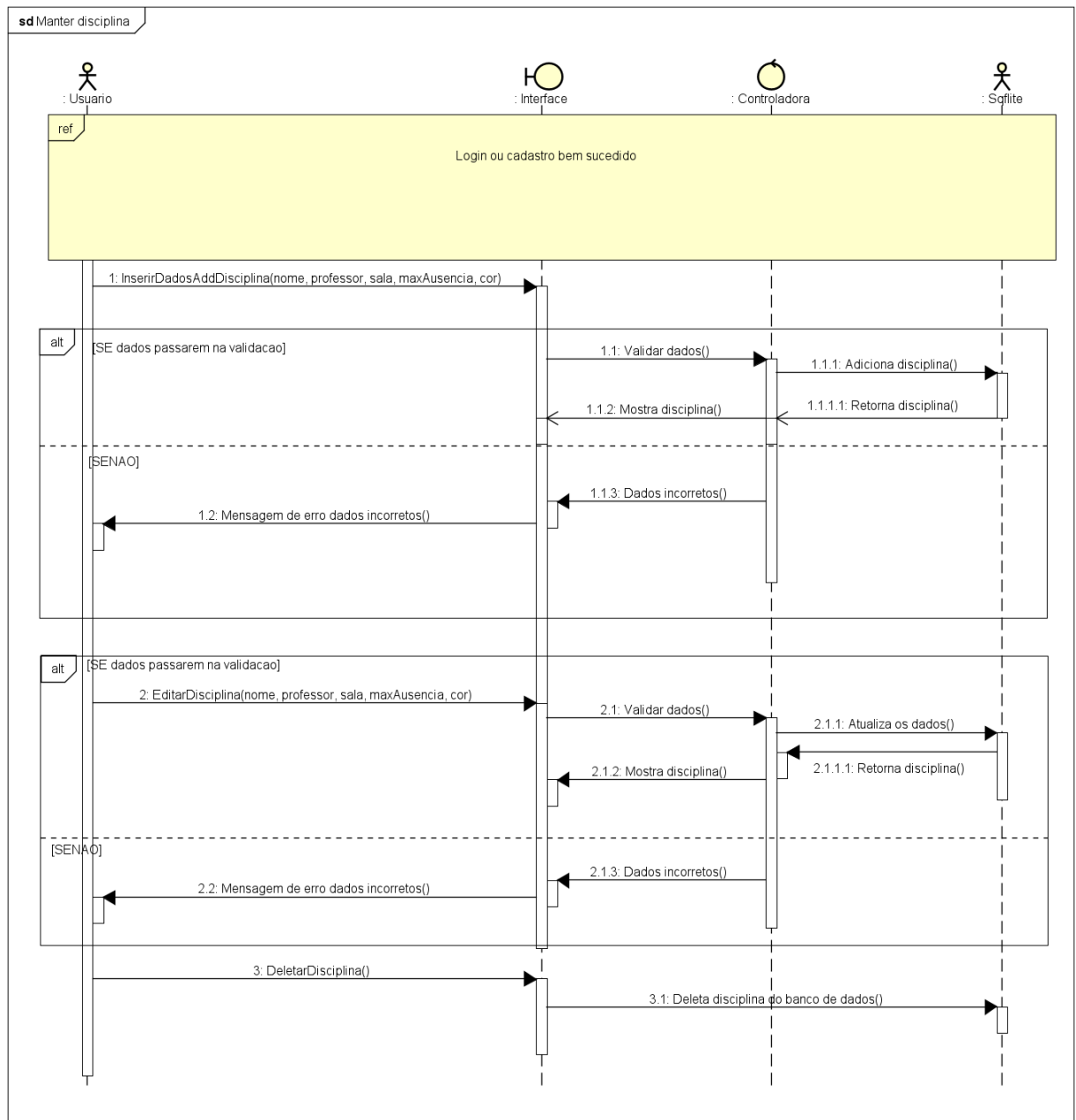
**Figura 58 – Diagrama de sequência do cadastro e login do usuário**



Fonte: Elaborado pelos autores .

A Figura 59 apresenta o diagrama de sequência de manter disciplina.

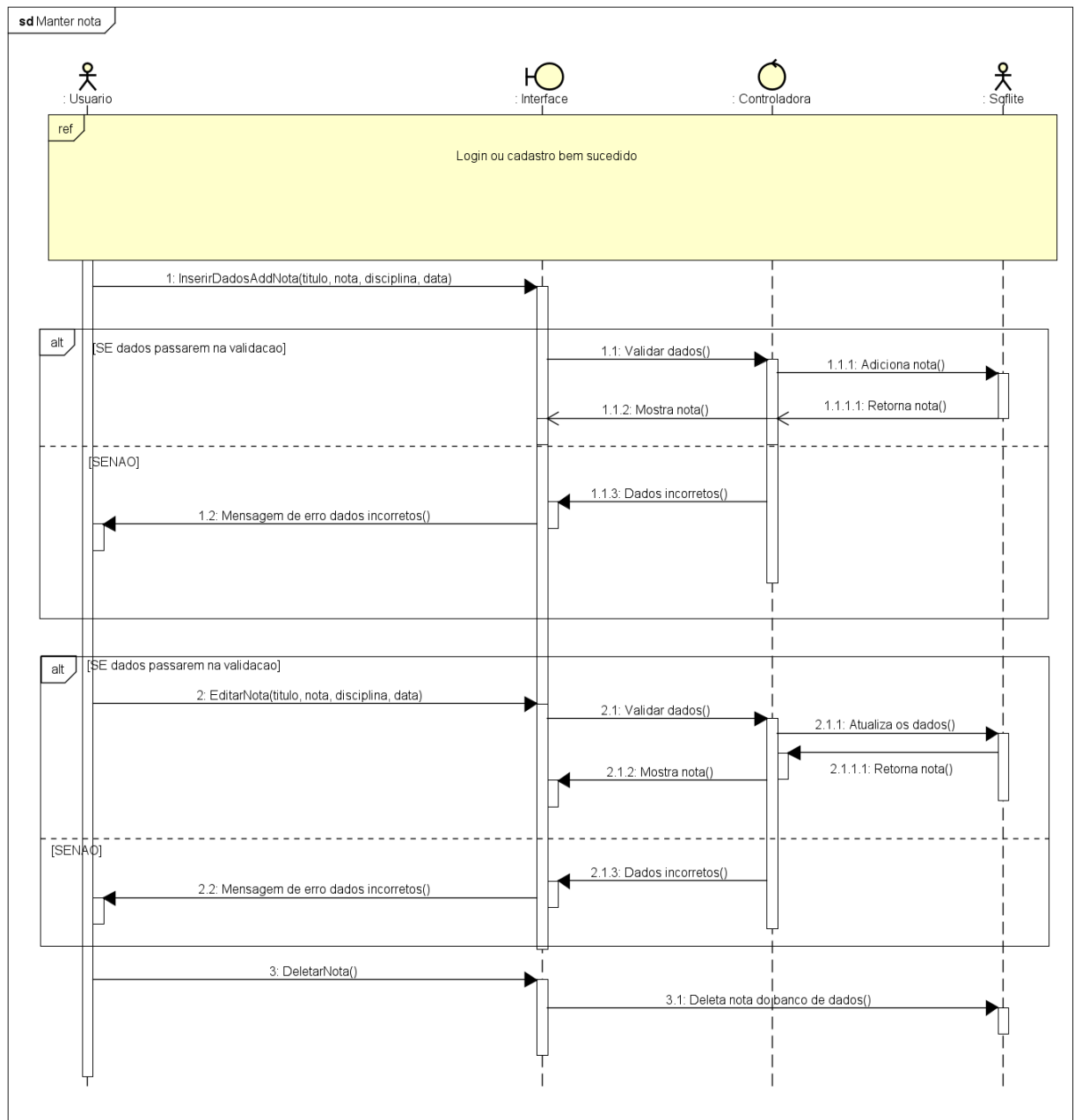
**Figura 59 – Diagrama de sequência de manter disciplina**



**Fonte: Elaborado pelos autores.**

A Figura 60 apresenta o diagrama de sequência de manter nota.

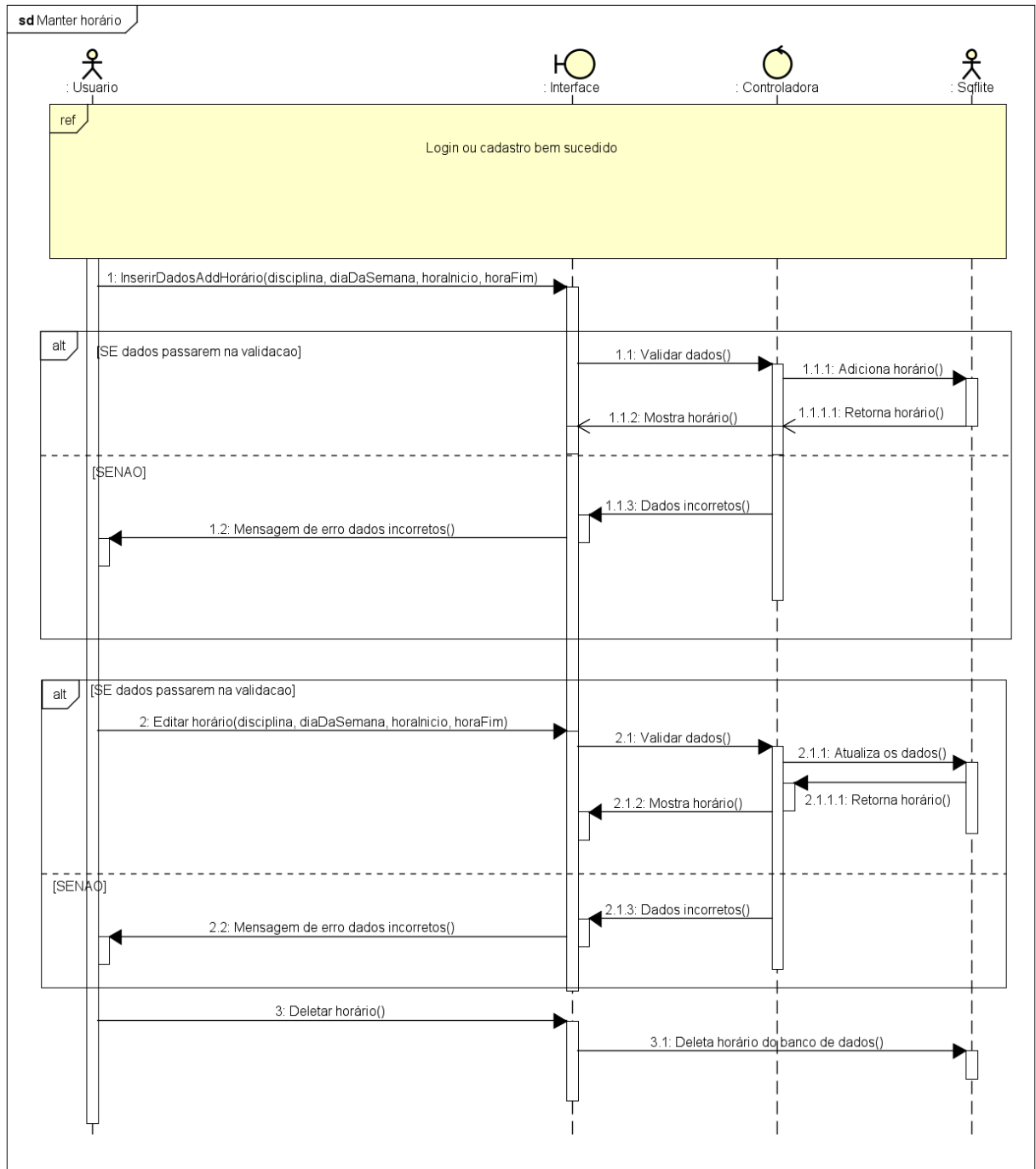
**Figura 60 – Diagrama a de Sequência de manter nota**



**Fonte: Elaborado pelos autores.**

A Figura 61 apresenta o diagrama de sequência de manter horário.

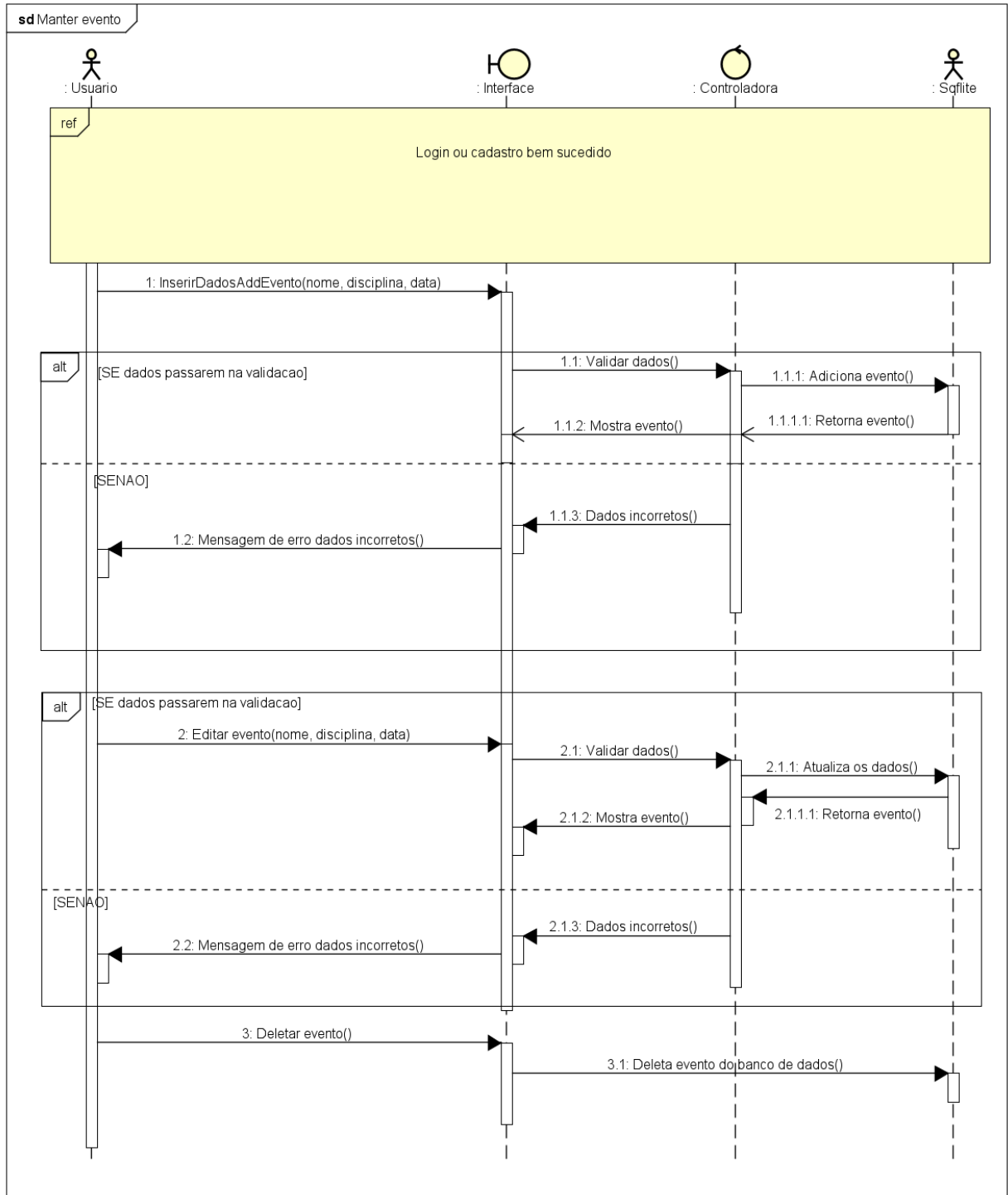
**Figura 61 – Diagrama de Sequência de manter horário**



**Fonte: Elaborado pelos autores .**

A Figura 62 apresenta o diagrama de sequência de manter evento.

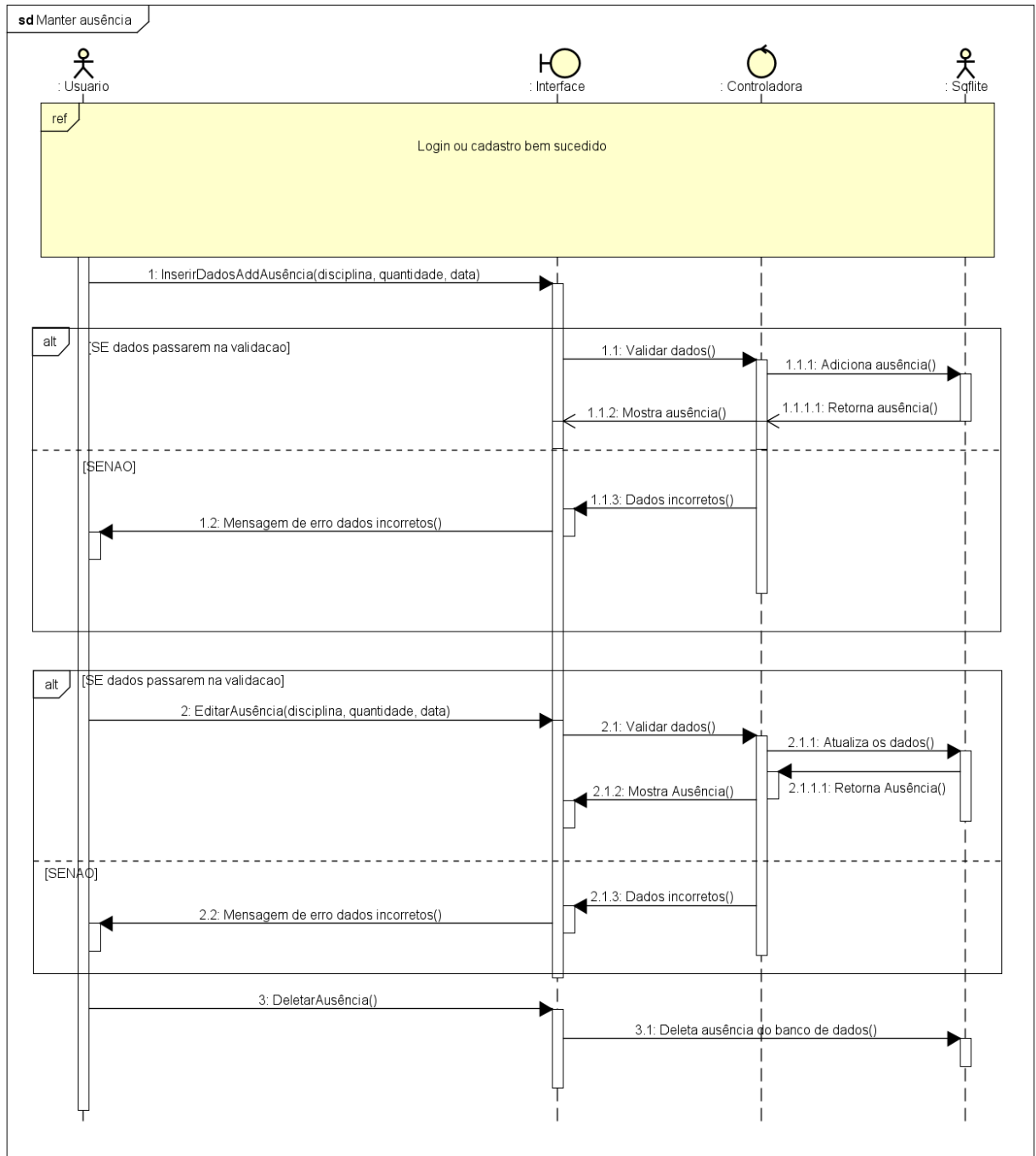
**Figura 62 – Diagrama de Sequência de manter evento**



Fonte: Elaborado pelos autores .

A Figura 63 apresenta o diagrama de sequência de manter ausência.

**Figura 63 – Diagrama de Sequência de manter ausência**



Fonte: Elaborado pelos autores .

#### 4.4. Banco de Dados

Segundo Oracle (2019), um banco de dados são várias informações aglomeradas em uma coleção estruturada, tipicamente armazenadas em um sistema



de computador, posteriormente em um servidor. Geralmente, banco de dados são controlados por um Sistema Gerenciador de Banco de Dados (SGBD).

Dados são modelados em linhas e colunas em várias tabelas associadas a eles, podem ser acessados, modificados, adicionados e excluídos através de consultas (*query*), como *SELECT*, *INSERT*, *UPDATE*, *DELETE*, dentre outras milhares de consultas.

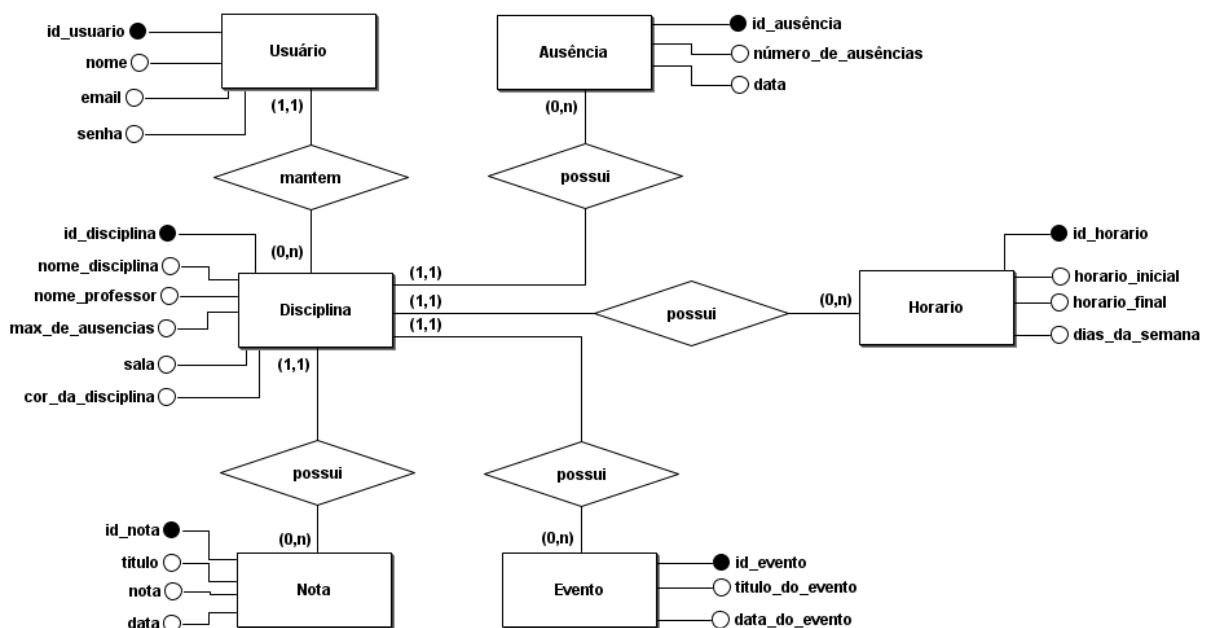
A grande maioria dos bancos de dados atualmente, utilizam um modelo estruturado de consulta, *Structured Query Language* (SQL), há alguns, no entanto, que utilizam a consulta por *Hibernate Query Language*, tópico mais avançado a ser discutido.

#### 4.5. Diagrama de Entidade e Relacionamento (DER)

Diagrama Entidade Relacionamento (DER) é um modelo diagramático que descreve o modelo de dados de um sistema com alto nível de abstração. Ele é a principal representação do Modelo de Entidades e Relacionamentos. Sua maior aplicação é para visualizar o relacionamento entre tabelas de um banco de dados, no qual as relações são construídas através da associação de um ou mais atributos destas tabelas.

A Figura 64 apresenta o DER do sistema.

Figura 64 - Representação o Diagrama de Entidade e Relacionamento do aplicativo



Fonte: Elaborado pelos autores

#### 4.6. Estruturação SQL

Para o armazenamento de dados do aplicativo utilizamos uma biblioteca de linguagem C que implementa um mecanismo de banco de dados SQL. O *SQLite* é um sistema gerenciador de banco de dados relacional que utiliza a linguagem SQL, na aplicação desenvolvida neste projeto os dados são armazenados localmente através do *SQLite*, dispensando o uso de um servidor. O *SQLite* é uma biblioteca compacta, com todos os recursos ativados o tamanho pode ser inferior a 750KB. Sobre o *SQLite*, pode-se afirmar que:

*SQLite* é uma biblioteca em processo que implementa um mecanismo de banco de dados SQL transacional, sem servidor, sem configuração e independente. O código para *SQLite* é de domínio público e, portanto, é gratuito para uso para qualquer finalidade, comercial ou privada. O *SQLite* é o banco de dados mais implantado no mundo, com mais aplicativos do que podemos contar, incluindo vários projetos de alto nível. (*SQLite*, 2021)

Para se utilizar o *SQLite* em aplicações *Flutter* é necessário a utilização de um pacote, o *SQFlite*, este *plugin* oferece os seguintes recursos:

1. Suporta *iOS* e *Android*;
2. Transações de suporte e lotes;
3. Gerenciamento automático de versão durante a abertura;
4. *Helpers* para consultas de inserção, consulta, atualização e exclusão;
5. Operação de banco de dados executada em um *thread* de segundo plano no *iOS* e *Android*

A Figura 65 apresenta uma captura de tela do objeto *Disciplina* e dos campos atribuídos a ele.

**Figura 65 - Representação do objeto Disciplina**

```
{
  "id": "int",
  "nome_disciplina": "string",
  "cor": "string",
  "max_ausencias": "int",
  "nome_professor": "string",
  "sala": "string",
  "nota": {
    "id": "int",
    "disciplina_id": "int",
    "titulo": "string",
    "nota": "float",
    "data": "datetime"
  },
  "horario": {
    "id": "int",
    "disciplina_id": "int",
    "hora_inicial": "datetime",
    "hora_final": "datetime",
    "dias_da_semana": "datetime"
  },
  "evento": {
    "id": "int",
    "disciplina_id": "int",
    "titulo_do_evento": "string",
    "data_do_evento": "datetime"
  },
  "ausencia": {
    "id": "int",
    "disciplina_id": "int",
    "num_ausencias": "int",
    "dia_ausencia": "datetime"
  }
}
```

Fonte: Elaborado pelos autores.

#### 4.7. Dicionário de Dados

O Dicionário de Dados (DD) consiste em uma lista organizada de todos os elementos de dados que são pertinentes para o sistema. As tabelas devem conter os seguintes campos:

**Entidade:** é o nome da entidade que foi definida no DER. A entidade é uma pessoa, objeto ou lugar que será considerada como objeto pelo qual temos interesse em guardar informações a seu respeito.

**Atributo:** Os atributos são as características da entidade cliente que desejamos guardar.

**Classe:** as classes podem ser: simples, composto, multivalorado e determinante. Simples indica um atributo normalmente. Composto indica que ele poderá ser dividido em outros atributos, como por exemplo, o endereço. Multivalorado é quando o valor do atributo poderá não ser único e determinante é um atributo que será usado como chave, como Cadastro de Pessoa Física (CPF), Código do cliente, ID do usuário.

**Domínio:** podem ser numéricos, texto, data, booleano ou map. Podemos chamar também de tipo do valor que o atributo irá receber. A definição desses tipos deve seguir um processo lógico, exemplo: nome é texto, salário é numérico, data de nascimento é data e assim por diante.

**Tamanho:** define a quantidade de caracteres que serão necessários para armazenar o seu conteúdo. Geralmente o tamanho é definido apenas para atributos de domínio texto.

**Descrição:** é opcional e pode ser usado para descrever o que é aquele atributo ou dar informações adicionais que possam ser usadas futuramente pelo analista ou programador do sistema.

As Tabelas 24 a 29 apresentam o Dicionário de Dados de cada entidade.

A tabela 24 apresenta o dicionário de dados da entidade usuário com todas os atributos, classes, domínios, tamanhos de cada propriedades e a descrição.

**Tabela 24 - Dicionário de dados da entidade usuário**

Entidade: USUÁRIO
-------------------

Atributo	Classe	Domínio	Tamanho	Descrição
ID_USUARIO	Determinante	Texto	30	Chave Primária da tabela usuário.
E-MAIL	Simples	Texto	50	E-mail do usuário
SENHA	Simples	Texto	20	Senha do usuário

Fonte: Elaborado pelos autores.

A tabela 25 apresenta o dicionário de dados da entidade disciplina com todas os atributos, classes, domínios, tamanhos de cada propriedades e a descrição.

**Tabela 25 - Dicionário de dados da entidade disciplina**

Entidade: DISCIPLINA				
Atributo	Classe	Domínio	Tamanho	Descrição
ID_DISCIPLINA	Determinante	Texto	20	Chave Primária da disciplina dentro do <i>document</i> .
NOME	Simples	Texto	20	Nome da disciplina
PROFESSOR	Simples	Texto	20	Nome do professor da disciplina
SALA	Simples	Texto	10	Nome da sala da disciplina
MAX_AUSÊNCIA	Simples	Numérico	100	Número máximo de ausências da disciplina

NOTAS	Composto	Array	-	Lista das Notas da disciplina
AUSÊNCIAS	Composto	Array	-	Lista das Ausências da disciplina
EVENTOS	Composto	Array	-	Lista dos Eventos da Disciplina

**Fonte: Elaborado pelos autores.**

A tabela 26 apresenta o dicionário de dados da entidade ausência com todas os atributos, classes, domínios, tamanhos de cada propriedades e a descrição.

**Tabela 26 - Dicionário de dados da entidade ausência**

Entidade: AUSÊNCIA				
Atributo	Classe	Domínio	Tamanho	Descrição
ID_AUSÊNCIA	Determinante	Texto	40	Index da Ausência dentro do <i>array</i> da disciplina
QUANTIDADE	Simple	Texto	20	Quantidade de ausência.
DATA	Simple	Texto	10	Data na qual a ausência ocorreu

**Fonte: Elaborado pelos autores.**

A tabela 27 apresenta o dicionário de dados da entidade nota com todas os atributos, classes, domínios, tamanhos de cada propriedades e a descrição.

Tabela 27 - Dicionário de dados da entidade nota

Entidade: NOTA				
Atributo	Classe	Domínio	Tamanho	Descrição
ID_NOTA	Determinante	Texto	40	Index da Nota dentro do <i>array</i> da disciplina
TÍTULO	Simples	Texto	20	Título da nota
VALOR	Simples	Texto	20	Quantidade de ausência.
DATA	Simples	Texto	10	Data na qual a ausência ocorreu

Fonte: Elaborado pelos autores.

A tabela 28 apresenta o dicionário de dados da entidade evento com todas os atributos, classes, domínios, tamanhos de cada propriedades e a descrição.

Tabela 28- Dicionário de dados da entidade evento

Entidade: EVENTO				
Atributo	Classe	Domínio	Tamanho	Descrição
ID_EVENTO	Determinante	Texto	40	Index do Evento dentro do <i>array</i> da disciplina
TITULO	Simples	Texto	20	Título do evento
DATA	Simples	Data	20	Data de termino do evento

Fonte: Elaborado pelos autores.

A tabela 29 apresenta o dicionário de dados da entidade horário com todas os atributos, classes, domínios, tamanhos de cada propriedades e a descrição.

**Tabela 29 - Dicionário de dados da entidade Horário**

Entidade: HORÁRIO				
Atributo	Classe	Domínio	Tamanho	Descrição
ID_HORÁRIO	Determinante	Texto	40	Index do Horário dentro do <i>array</i> da disciplina
HORÁRIO_INICIAL	Simple	Data	20	Hora inicial da aula
HORÁRIO_FINAL	Simple	Data	20	Hora final da aula
DIAS_DA_SEMANA	Simple	Data	10	Dias da semana em que a aula acontece

**Fonte: Elaborado pelos autores.**



## 5. CONSIDERAÇÕES FINAIS

Diante dos diversos problemas acadêmicos que ocorreram com os estudantes durante a pandemia e as consequências que vieram a seguir, como: os resultados do Sistema de Avaliação de Rendimento Escolar do Estado de São Paulo, no qual os alunos do ensino médio tiveram o pior desempenho da história e também que 67% dos estudantes do ensino médio se queixaram de dificuldades em estabelecer uma organização diária de estudos, teve-se a ideia para desenvolver o aplicativo *School Schedule*, com a finalidade de auxiliar estudante a se organizarem melhor na vida acadêmica.

Ao decorrer do desenvolvimento foi criado o protótipo da aplicação que auxiliaria os alunos nesses problemas, se foi necessário estudar todos os fluxos que seriam implementados na aplicação, modelagem do banco de dados, prototipagem da interface, desenvolvimento e estudo dos casos de uso e diversos outros tipos de diagramas que foram necessários para guiar o desenvolvimento da aplicação. O processo de desenvolvimento do aplicativo foi realizado seguindo todos os padrões estabelecidos no planejamento do projeto desde diagramas, fluxos, modelagem do banco de dados, protótipos, ferramentas e todos os requisitos funcionais e não funcionais, aplicando a metodologia *scrum* foi possível gerenciar o projeto de forma eficiente, estabelecendo etapas e metas para equipe de acordo com as necessidades e tempo previsto para entrega do projeto.

Com o desenvolvimento do projeto finalizado, ficou evidente que se utilizar um aplicativo para gerenciar sua vida acadêmica auxilia na organização dos estudos, tornando-se possível manter suas disciplinas, notas, ausências, eventos e também ficar sabendo os dias e horários das aulas. Sendo assim, o aplicativo traz uma possível solução para estudantes terem mais controle do seu desempenho nos estudos.

A partir do processo de desenvolvimento foi possível identificar diversos pontos de melhorias que serão implementados futuramente no projeto, dentre eles: adicionar modo escuro como o aplicativo é majoritariamente branco é fundamental que tenha um tema escuro para atrair os usuários que tem preferência por esse tema; Adicionar uma opção de notificação com personalização de tempo, por exemplo ser notificado 1 dia, 1 hora ou 10 minutos antes de uma aula ou evento começar; Adicionar outros idiomas para abranger usuários de outras nacionalidades e publicar o aplicativo na

loja de apps *Google Play Store* com o objetivo de alcançar mais usuários e trazer mais disponibilidade.

## REFERÊNCIAS

Agenda do universitário. Disponível em:

[https://play.google.com/store/apps/details?id=guipfernandes.github.com.agenda\\_do\\_universitario&hl=en\\_US&gl=US](https://play.google.com/store/apps/details?id=guipfernandes.github.com.agenda_do_universitario&hl=en_US&gl=US). Acesso em: 9 de mar. 2022, às 7h57min.

Agenda *iScholar*. Disponível em:

[https://play.google.com/store/apps/details?id=com.claudivan.agendadoestudanteplus&hl=pt\\_BR&gl=US&pli=1](https://play.google.com/store/apps/details?id=com.claudivan.agendadoestudanteplus&hl=pt_BR&gl=US&pli=1). Acesso em: 9 de mar. 2022, às 7h55min.

*ANDROID STUDIO*. Disponível em: <https://developer.android.com/studio>. Acesso em: 10 de mar. 2022, às 8h10min.

*Dart* documentação. Disponível em: <https://dart.dev>. Acesso em: 10 de mar. 2022, às 8h27min.

*Flutter* documentação. Disponível em: <https://flutter.dev>. Acesso em: 10 de mar. 2022, às 8h29min.

*Firebase* documentação. Disponível em: <https://firebase.google.com/?hl=pt> Acesso em: 10 de mar. 2022, às 8h47min.

*Microsoft*. Disponível em: <https://visualstudio.microsoft.com/> Acesso em: 11 de mar. 2022, às 7h30min.

Nota U. Disponível em:

[https://play.google.com/store/apps/details?id=com.notasu&hl=pt\\_BR&gl=US](https://play.google.com/store/apps/details?id=com.notasu&hl=pt_BR&gl=US). Acesso em: 9 de mar. 2022, às 8h05min.

*PUB.DEV* plugins para *flutter*. Disponível em: <https://pub.dev/packages/sqlite> Acesso em: 12 de mar. 2022, às 4h43min.

OKUMURA, Renata. **Pesquisa mostra que 67% dos alunos têm dificuldade para organizar estudos online na pandemia**. Estadão. 2020. Disponível em: [http://www.abed.org.br/arquivos/Pesquisa\\_mostra\\_que\\_67\\_por\\_cento\\_dos\\_alunos\\_tem\\_dificuldade.pdf](http://www.abed.org.br/arquivos/Pesquisa_mostra_que_67_por_cento_dos_alunos_tem_dificuldade.pdf). Acesso em: 15 mar. 2022, às 15h34min.

SOMMERVILLE, I. **Engenharia de Software**. 8ª Edição. Editora: Pearson Addison-Wesley. São Paulo, 2007. Acesso em: 10 de mar. 2022, às 8h14min.

VIANA, Daniel. **“Firebase: descubra no que esta plataforma pode te ajudar”**. TreinaWeb Tecnologia LTDA 2004 – 2019. Disponível em: <https://www.treinaweb.com.br/blog/firebase-descubra-no-que-esta-plataforma-pode-te-ajudar/>. Acesso em: 26 mar. 2022, às 15h20min.

VIEIRA, Bárbara. **Com pandemia, aluno do ensino médio de SP tem pior desempenho da história; estudante sai com defasagem de seis anos**. G1. 2022. Disponível em: <https://g1.globo.com/sp/sao-paulo/educacao/noticia/2022/03/02/pandemia-afeta-aprendizagem-e-966percent->

dos-alunos-deixam-ensino-medio-em-sp-com-nivel-abaixo-do-adequado-em-matematica.ghtml. Acesso em: 15 mar 2022, às 15h49min.

VIEIRA, Rodrigo. “**UML — Diagrama de Casos de Uso**”. *Operational TI*. 2019. Disponível em: <https://medium.com/operacionalti/uml-diagrama-de-casos-de-uso-29f4358ce4d5>. Acesso em: 11 jun. 2022, às 8h31min.